ORDINATEUR INDIVIDUEL



PHC 25

NOTICE D'EMPLOI

Vous avez fait le choix d'un des meilleurs micro-ordinateurs qui soient, le Sanyo PHC 25.

Nous vous félicitons, et sommes certains que vous tirerez de grandes satisfactions de votre appareil.

Cette notice d'emploi a pour but de vous expliquer l'utilisation du PHC 25.

Les instructions du Basic sont toutes commentées et agrémentées d'exemples qui vous permettront d'en apprécier les performances.

Toutefois, la puissance du langage Basic du PHC 25 nécessite un minimum de connaissances en programmation pour tirer le meilleur parti de votre ordinateur.

Pour celà, nous vous recommandons d'acquérir un des nombreux ouvrages d'initiation au Basic disponibles en librairie. Votre Revendeur Sanyo pourra vous conseiller utilement.

****

SOMMAIRE

MODE D'EMPLOI GENERAL

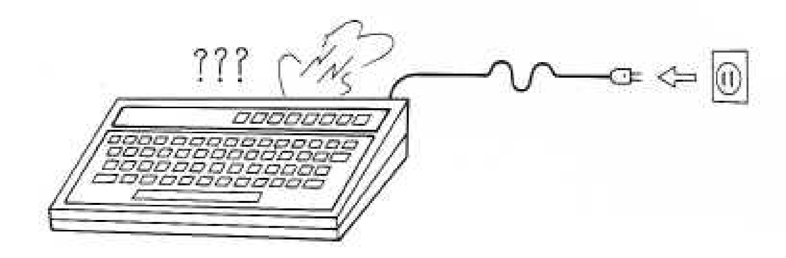
MODE D'EMPLOI DU BASIC

ANNEXES

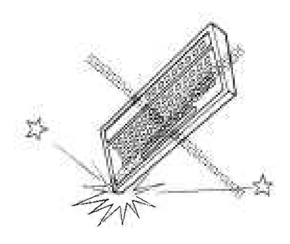
I. MODE D'EMPLOI

I.1. Précautions :

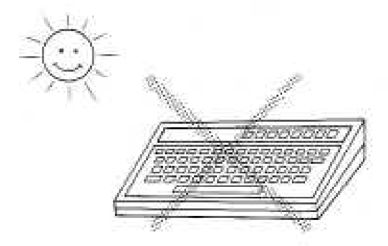
Débrancher l'interrupteur au moindre symptôme anormal et contacter votre distributeur SANYO.



Evitez les chocs, ne laisser pas tomber votre PHC 25



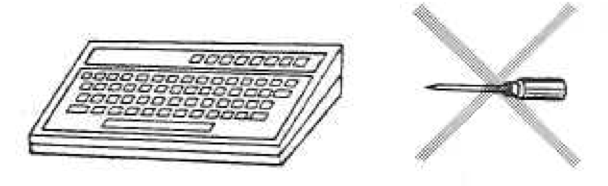
Evitez d'employer l'appareil dans un endroit chaud, humide ou exposé aux rayons directs du soleil.



Evitez de verser du liquide sur l'appareil (café, etc. . . .)



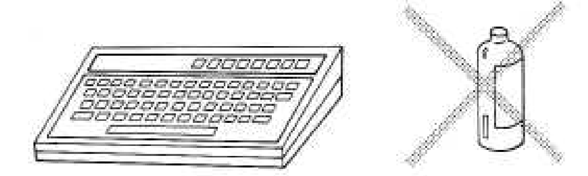
Attention, certains éléments peuvent être la cause d'électrocution. Ne jamais ouvrir l'appareil sous tension, la garantie ne pouvant être assurée sur un appareil ayant été ouvert.



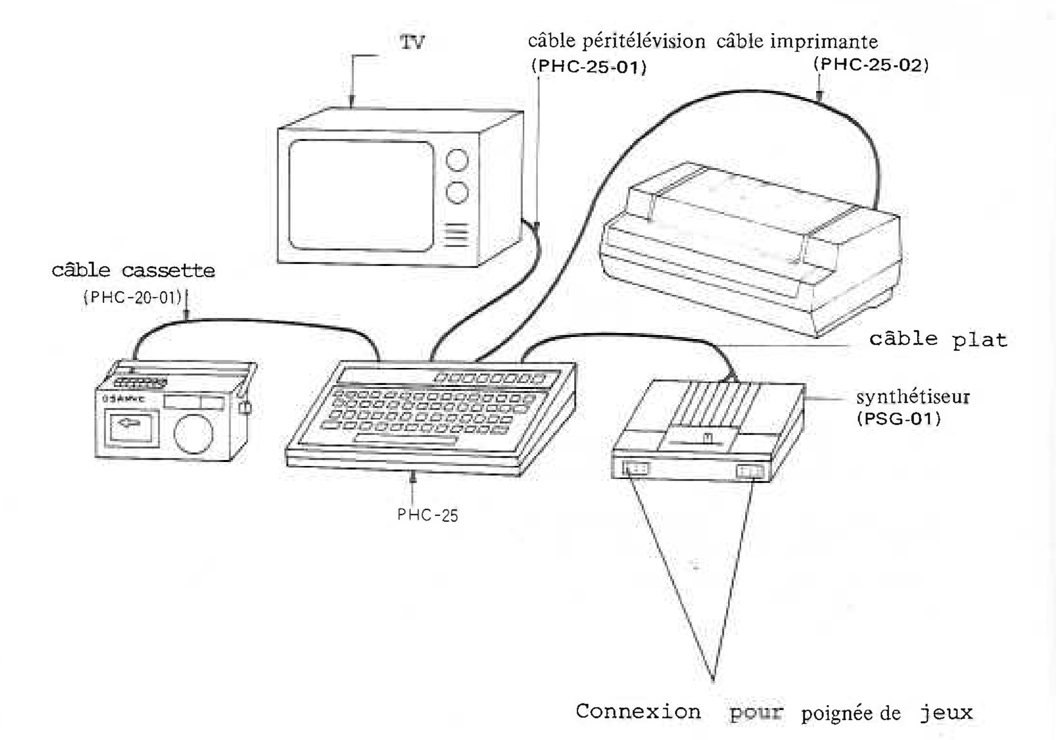
Ne pas gêner les ventilations prévues. Ne jamais insérer d'objets métalliques ou inflammables.

Utiliser un chiffon sec pour le nettoyage de l'appareil et n'employer aucun produit chimique.

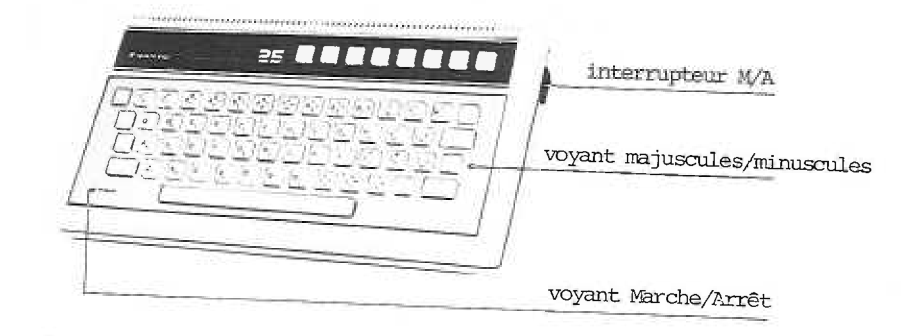
N'oubliez pas d'éteindre l'appareil lorsqu'il n'est pas en service.

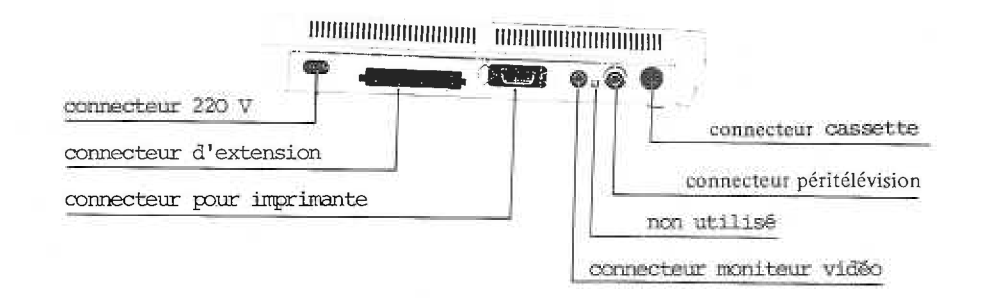


I.2 LE PHC 25 et ses périphériques :

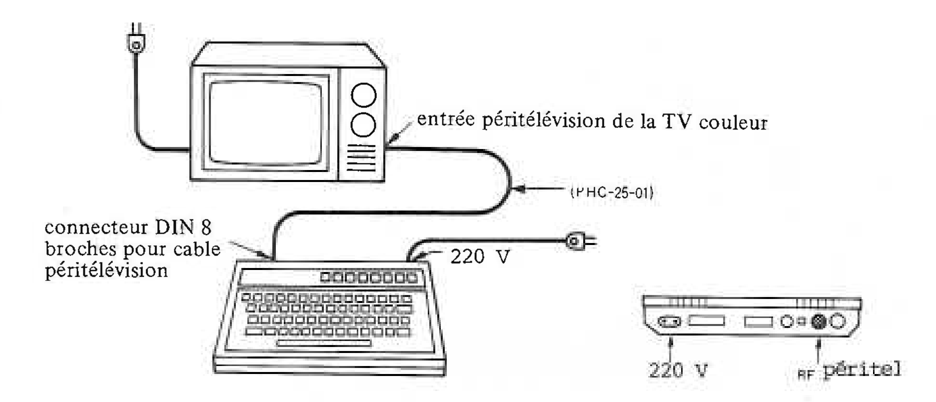


I.3. Nomenclature :





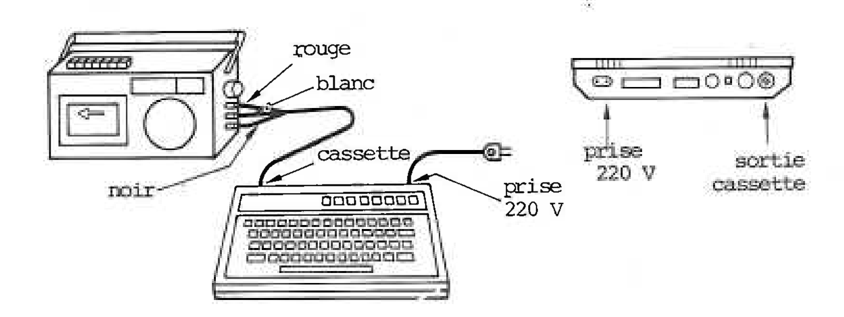
I.4. Connection avec les périphériques :



Branchement du magnétophone.

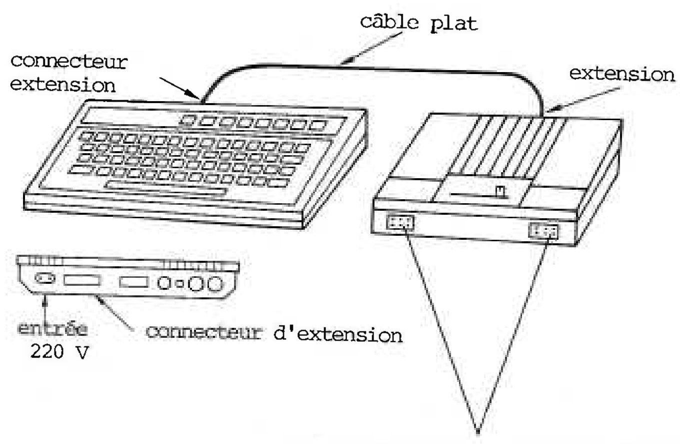
Vérifier que votre câble de connection est bien branché :
- Borne rouge sur la position "MIC" (microphone)
- Borne blanche "EAR" (écouteur)
- Borne noire (commande à distance) doit être branché sur la position REM, sauf dans le cas où votre magnétocassette ne possède pas de commande à distance.

Votre magnétocassette doit être équipé de prises standard femelles : diamètre 3,5 mn pour EAR et MIC
: diamètre 2,5 mn pour REM.



I.6 Connection avec le synthétiseur musical :

Il existe un synthétiseur musical (PSG 01) qui peut être connecté au bus d'extension prévu à l'arrière de l'appareil.



Sur le synthétiseur PSG 01, vous pouvez brancher des manettes de jeux (JOYSTICK).

II. MISE EN SERVICE DU PHC 25 :

Brancher et allumer d'abord l'équipement périphérique (Moniteur Vidéo SG 12, poste de TV ou magnétocassette) avant d'allumer le PHC 25.

Lorsque vous allumez le PHC 25,

SCREEN ?

doit s'afficher en haut de l'écran, sur le côté gauche.

Répondez en composant 1 ou 2.
L'écran doit afficher :

```
SCREEN ? 2 (ou 1)
PHC 25 SANYO BASIC V1. 3
User   Memory   8121   Bytes   (ou   14265)
Ready
```

indiquant que l'appareil est prêt à fonctionner.

Pourquoi Screen 1 ou 2 ?

On peut utiliser le PHC 25 avec 1 ou 2 pages d'écran. Ceci est très utile lorsqu'on veut faire des calculs sur 1 page écran pendant qu'on affiche la deuxième page. Dans le mode Screen 1, sur une seule page utilisée la mémoire disponible est de 14265 caractères. Dans le cas où l'on utilise 2 pages écran, il reste 8121 caractères pour le programme. On peut changer à l'aide de l'instruction Screen dont vous trouverez la description plus loin, le nombre de pages-écran choisi au départ. Dans le cas où un programme à été écrit en utilisant les 2 pages écran, le fait de n'en selecter qu'une provoque des erreurs et le message "illegal function call"

ARRET

L'arrêt du poste TV avant celui du PHC 25 peut provoquer des anomalies sur votre téléviseur. Il faut donc d'abord enclencher l'interrupteur (OFF) du PHC 25, ensuite fermer les périphériques (TV). L'extinction de l'appareil provoque l'effaçage de tous les programmes stockés. Il faut donc les sauvegarder sur cassettes.

N.B. Il est indispensable d'attendre plus de 5 secondes avant de rebrancher le PHC 25 afin d'éviter des ennuis techniques. Il est interdit de fermer le poste TV avant la fermeture du PHC 25.

III. UTILISATION DU MAGNETOPHONE A CASSETTES

Les programmes peuvent être sauvegardés sur des cassettes standard. (Audio) beaucoup moins chères que les cassettes informatiques. Néanmois dans le cas d'utilisation intensive de cassettes et en particulier sur des applications professionnelles, nous vous conseillons d'utiliser une cassette informatique qui a une durée de vie et une fiabilité bien supérieure.

1) pour la connection (voir page 6)

2) règlage du niveau de son :
Mettre le niveau du volume presque à son maximum et le niveau de tonalité (s'il existe) vers le haut (aigu). Si le PHC 25 ne sauvegarde pas, n'écrit pas ou ne lit pas, il faut ajuster le niveau du volume et de la tonalité.

3) pour sauvegarder des programmes sur cassettes, il faut :

* introduire une cassette vierge dans le magnétocassette.

* composer au clavier CSAVE"ABC"

ABC étant le nom d'un programme pris comme exemple et ne devant pas avoir plus de huit caractères.

* Appuyer sur la touche RECORD et PLAY du magnétocassette en même temps, et ensuite Appuyer sur la touche RETURN du clavier du PHC 25.

A ce moment le programme en mémoire dans le PHC 25 sera en période de sauvegarde sur la cassette. La sauvegarde (l'enregistrement d'un programme) est terminé lorsque le message READY apparait à l'écran.

ATTENTION: Si vous sauvegardez un programme sur une cassette déjà enregistrée, vérifiez que la sauvegarde est faite à un endroit de la cassette vierge faute de quoi vous effaceriez le premier enregistrement de la cassette.

Le défilement de la cassette s'arrête si votre appareil possède la prise de télécommande REMOTE. Sinon, l'arrêter manuellement. Il est utile de noter le chiffre au compteur. La méthode de vérification est décrite plus bas paragr. d).

Pour lire le programme enregistré sur la cassette :

a) insérer la cassette et la remettre à l'endroit noté par le compteur,

b) composer CLOAD"ABC" (ABC étant l'exemple d'un nom de programme)

c) appuyer sur la touche RETURN du PHC 25 puis aussitôt sur la touche PLAY du magnétocassette.

Aussitôt que le programme est trouvé :

Found ABC apparaitra à l'écran.

Il faut attendre que le PHC 25 ait fini le chargement de ce programme et voir : "Ready" s'afficher à l'écran.

Le défilement s'arrête si votre appareil possède la prise de télécommande REMOTE. Sinon l'arrêter manuellement.

NOTE

Lorsque Skip suivi d'un programme "XXX" est affiché, c'est que le programme XXX se trouve avant celui que vous recherchez, la recherche continue.

d) pour vérifier si un programme a été correctement sauvegardé, il faut :

entrer CLOAD?"ABC"(nom d'un exemple de programme)

enclencher la touche PLAY de votre magnétocassette suivi de la touche RETURN du PHC 25. Si vous obtenez la réponse :

Ready

sur votre écran, le programme a bien été enregistré.

Autrement :

Bad

s'affichera et il vous faut recommencer la sauvegarde une deuxième fois.

La commande CLOAD? permet de comparer le contenu de la mémoire du PHC 25 avec le contenu de la partie de la cassette que vous êtes en train de lire.

IV. LE CLAVIER

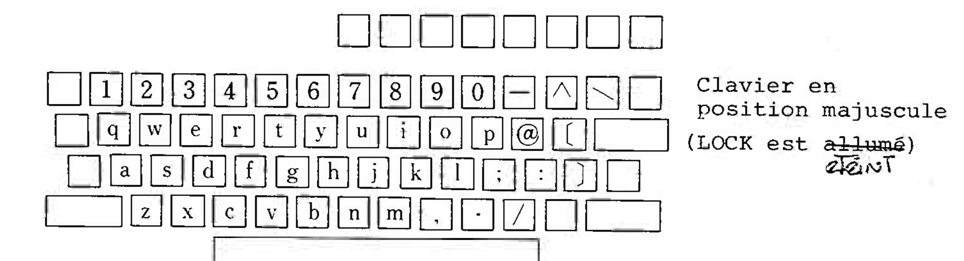
Les touches qui sont blanches sur les schémas sont appelées "touches générales".

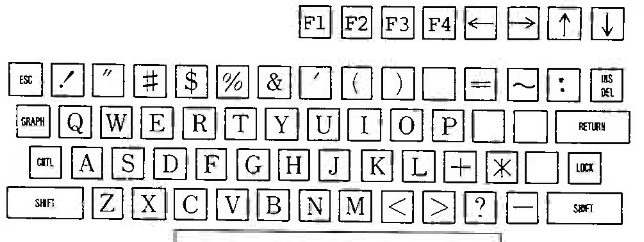
Lorsque vous appuyez sur une touche a par exemple, le caractère a apparait à l'écran

Ceci est valable pour toutes les lettres minuscules.

Les majuscules s'obtiennent en appuyant sur la touche SHIFT ou la touche LOCK.

La barre centrale s'appelle la barre d'espacement.



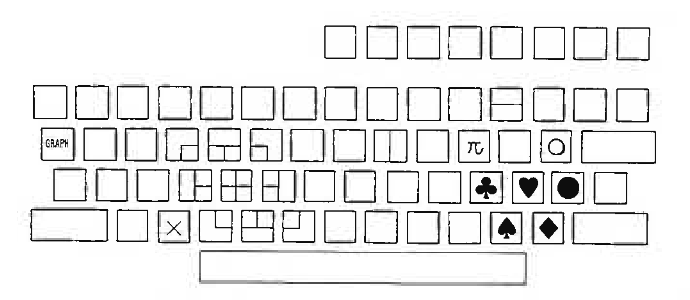


Clavier en position majuscule (LOCK est allumé), et en appuyant sur la touche SHIFT en même temps.

V. TOUCHES GRAPHIQUES

Lorsque vous appuyez sur la touche GRAPH et en même temps sur une touche figurant sur la table graphique ci-dessous, vous obtiendrez une frappe graphique.

Pour avoir les symboles graphiques, LOCK doit être éteint.



On appelle curseur le carré de lumière qui clignote sur l'écran pour vous indiquer votre position. Ce curseur bouge à la ligne suivante, lorsqu'une ligne est pleine et lorsqu'elle arrive en fin de page, l'écran monte automatiquement une nouvelle ligne prête à l'inscription.

Les flêches à droite du clavier → ← ↑ ↓

permettent au curseur de bouger à n importe quelle position

AUTO-REPEAT (répétition automatique)

Cette fonction de répétition existe pour toutes les touches.
Il suffit de maintenir une touche plus d'une seconde.
Cette possibilité existe en majuscule, en minuscule et en graphique, sauf pour les touches de fonctions F1, F2, F3, F4.

CLS

Effaçage complet de l'écran - en composant CLS sur les touches du clavier puis en appuyant sur RETURN

Correction partielle d'un ou plusieurs caractères

Exemple : Au lieu de taper DUPONT vous avez fait une erreur de frappe et tapé DURONT. Il suffit de remettre le curseur sur l'emplacement du R de DURONT et de retaper un P pour corriger et obtenir DUPONT. Dans le cas d'une lettre en trop, comme dans l'exemple DUPOONT, il suffit de se mettre sur un O et d'utiliser la touche INS/DEL pour effacer la lettre "O" en trop.

De même, lorsqu'il manque une lettre, vous pouvez l'insérer en utilisant les touches SHIFT INS/DEL simultanément et retaper avec l'orthographe correcte.

TOUCHE RETURN

La touche orange RETURN sert à confirmer une commande.

Les caractères affichés sur l'écran doivent être confirmés par une pression sur la touche RETURN.

ESC Touche Escape

Cette touche sert à compléter une commande et à se tenir prêt pour la commande suivante.

Lorsque l'on utilise ESC on arrête le deroulement du programme du PHC 25 temporairement.

Ces opérations sont reprises lorsqu'on appuie une nouvelle fois sur la touche ESC.

L'emploi de la touche ESC et CTRL simultanément permet d'arrêter un enregistrement ou une lecture de programme sur cassette à condition que le PHC 25 ait trouvé un programme sur la cassette (même un autre que celui appelé.

Exemple : Vous insérez une cassette programmée dans votre magnétocassette et vous lancez une procédure de lecture en faisant une faute de frappe dans le libellé du programme, il est possible d'arrêter le défilement de la cassette en appuyant simultanément sur ESC.CTRL puis de retaper correctement le nom du programme.

Touches Fonction

Sur le haut du clavier il y a 4 touches fonction :

F1..............RUN+RETURN	SHIFT F1........COLOR
F2 .... CLOAD"	SHIFT F2........CSAVE"
F3 .... PRINT	SHIFT F3........INPUT
F4 .... LIST	SHIFT F4........SCREEN

Ces touches sont reprogrammables, voir la fonction KEY.

UTILISATION DES PROGRAMMES DE JEUX

ENREGISTRES SUR CASSETTES

1.- . S'assurer que vous utilisez bien un magnétocassette SANYO TRC 1550 ou équivalent.

. La vitesse de défilement doit être constante.

. Il est recommandé par conséquent d'utiliser un branchement sur secteur 220 V (par l'intermédiaire de l'adaptateur/chargeur 6 volts).

.L'inconvénient de l'utilisation de piles sèches est que la vitesse varie avec l'état des piles.

2.- . Règler la vitesse à 0 et le volume à 8.

3.- . Vérifiez que votre câble de connection est bien branché :

- . borne rouge sur la position MIC (microphome)
- . borne blanche " " EAR (écouteur)
- . borne noire " " " REM (commande à distance)

4.- . Entrez CLOAD "nom du programme" RETURN

(Les guillements s'obtiennent en appuyant sur la touche SHIFT et la touche 2 en même temps).

. Ensuite, appuyez sur la touche "PLAY" de votre magnétocassette pour le chargement du programme que vous avez sélectionné.

ATTENTION : L'ordinateur fait la différence entre les majuscules et les minuscules pour les noms de programme aussi il est necessaire de composer le nom comme il est indique au nom de code.

6.- . Aussitôt que le programme est trouvé, FOUND s'affiche avec le nom du programme.

exemple : found ABC

7.- . Il faut attendre que le programme soit entièrement chargé.

. A ce moment READY s'affiche sur l'écran et vous pouvez commencer votre jeu.

. Si BAD se trouve affiché, il faut reprendre à 5 et vérifier le contrôle de volume de votre magnétocassette.

****

EXEMPLE D'UTILISATION DE PROGRAMME

		No de code
KEYBOARD	: clavier	: "key"
BLOCK-BREAKING	: casse-briques	: "block"
SPACE TRAVEL 2	: Jeu de l'espace No2	: "space"
OTHELLO	: Othello	: "othelo"
PERPETUEL CALENDAR	: Calendrier de l'an 1 à l'an 2000	: "calen"
DEMO	: démonstration des possibilités graphiques	: "demo"
INTRO	: présentation du PHC 25	: "intro"

================================

================================

N.B. Tous les jeux peuvent être arrêtés avec la touche ESCAPE (ESC) et redémarrent avec la même touche.

Dans le cas où l'on veut repartir du début du jeu , on appuie une fois sur la touche ESC puis sur la touche RETURN plusieurs fois jusqu'au moment où l'écran devient vide, en appuyant sur la touche F1 (RUN) le déroulement du programme reprend au début.

================================

KEYBOARD

Auto-enseignement visuel du clavier de machine à écrire.

Il s'agit d'un des meilleurs exemples d'auto-enseignement qui peut être employé à tous les âges à partir de 8 ans.

Suivant le niveau de vos connaissances, vous pouvez choisir le rythme qui vous convient le mieux :

1.- Très lent
2.- Lent
3.- Moyen
4.- Rapide
5.- Très rapide

Pour commencer :

- Appuyez sur la touche RETURN -

Le clavier apparaît sur l'écran.

Il s'agit de taper sur la touche correspondant à la lettre qui clignote sur l'écran.

Pour ce faire , vous disposez d'un temps limité.

Si vous appuyez sur la touche dans le temps voulu, vous gagnez des points.

Vos essais sont enregistrés.

Le SCORE maximum obtenu se trouve affiché.

(La rubrique "CHANCE" = Nombre de fois restant à jouer).

L'utilisateur peut gravir plusieurs échelons de vitesses pour arriver à une connaissance parfaite du clavier en même temps qu'à un bon entraînement de ses réflexes.

On peut également, si on le désire, améliorer les résultats en ajoutant un cache-clavier au-dessus de l'appareil pour apprendre la dactylographie sans regarder son clavier.

****

BLOCK-BREAKING

CASSE-BRIQUES code "block"

Jeu de démolition d'un mur par une balle

- Vous avez 3 balles pour chaque partie.

- Il faut chercher à faire rebondir la balle contre votre raquette pour qu'elle démolisse le mur.

- L'écran affiche le nombre de balles restant à jouer et le SCORE.

Mode d'emploi :

* Appuyez sur la touche RETURN

* Utilisez les touches [↑] [↓]

La direction de la balle peut être changée par le filet.

Le jeu est terminé lorqu'il n'y a plus de balles.

Le score s'affiche au milieu de l'écran et le meilleur score obtenu s'affiche à droite.

****

JEU DE L'ESPACE No 2

Code "space"

SPACE est un jeu d'adresse et de rapidité. Vous êtes pilote d'un vaisseau spacial et votre route est semée d'embûches.

Votre but est de percuter les bases ennemies o

(dans ce cas, vous gagnez 5 points)

ou les bases ennemies ♣

(dans ce cas vous gagnez 50 points minimum)

Vous devez éviter les " ! " qui sont des missiles ennemis

Si vous succombez à leur pouvoir destructeur, une croix (x) s'affiche en signe de deuil.

Mode d'emploi :

1) Appuyez sur la touche RETURN

2) Le vaisseau (V) apparaît sur l'écran et démarre

3) Utilisez les touches [→] et [←] pour commander votre vaisseau

4) La vitesse augmente au fur et à mesure que la partie s'avance.

5) Lorsque les trois vaisseaux sont détruits, "GAME OVER" (jeu terminé) s'affiche sur l'écran

Le score s'affiche sur l'écran en haut et à droite

****

OTHELLO

code "othelo"

Le jeu bien connu OTHELLO s'affiche sur l'écran. C'est un descendant direct du jeu REVERSI qui existe depuis 1870. On peut y jouer à partir de 10 ans.
Vous pouvez choisir entre trois possibilités :

1) Jouer contre une autre personne
2) Jouer contre l'ordinateur
3) Démonstration de jeu de l'ordinateur contre lui-même

On joue sur 64 cases qui sont numérotées :

HORIZONTALEMENT : ("X") 1 à 8
VERTICALEMENT : ("Y") 1 à 8

Les joueurs indiquent la position qu'ils ont choisie, toujours dans le même ordre : "X"d'abord, "Y" ensuite. Cette question leur est posée par l'ordinateur lorsque leur tour de jouer arrive.

Mode d'emploi :

1) contre un adversaire :
vos pions sont identifiés par le signe ●
et ceux de l'adversaire par le signe ○

2) contre l'ordinateur :
vos pions sont identifiés par le signe ●
ceux de l'ordinateur par le signe ○

Le but du jeu :

PRENDRE L'ADVERSAIRE DE FLANC c'est-à-dire de chaque côté de ses pions.

* Les joueurs indiquent chacun un pion à tour de rôle en s'efforçant d'entourer un ou plusieurs pions de l'adversaire situés côte à côte et en ligne droite, que ce soit horizontalement, verticalement ou en diagonale.
* Ils ont alors le droit de s'approprier les pions de l'adversaire entourés par les pions de leur camp et de transformer ainsi ces pions "pris" en leurs propres pions.
* Lorsque l'on débute, quatre pièces figurent déjà à l'écran.
* Vous commencez en indiquant la position horizontale "X" qui vous est demandée sur l'écran et en appuyant sur la touche RETURN * Ensuite, vous indiquez la position verticale "Y" qui vous est demandée en appuyant sur la touche RETURN.
- Si "ERROR" s'affiche, c'est que vous avez donné un numéro incorrect. Il suffit d'appuyer sur RETURN pour corriger votre réponse.
- Lorsque c'est au tour de l'ordinateur à jouer, il affiche "THINKING" ---- (temps de réflexion).
Le score s'affiche à droite de l'écran et note le nombre de pions qui vous restent.

****

CALENDRIER PERPETUEL

code "calen"

Il s'agit d'un calendrier perpétuel que vous pouvez afficher sur l'écran.

Vous avez le choix, à partir de l'an 1 de notre ère, jusqu'à l'an 2 000.

Vous pouvez, par exemple, demander ainsi le jour de la semaine de votre date de naissance. Le calendrier perpétuel répondra à votre question.

Mode d'emploi :

1) Appuyez sur la touche RETURN

2) L'ordinateur vous demande "YEAR" (année)

3) Composez les chiffres de l'année désirée et Appuyez sur RETURN

4) L'ordinateur vous demande "MONTH" (Mois)

5) Composez le/les chiffres du mois de 1 à 12 et Appuyez sur RETURN

Le calendrier demandé apparaîtra sur l'écran.

****

DEMO

Ce programme est automatique, bouclé sur lui-même, il montre les possibilités graphiques et couleurs du PHC 25.

INTRO

Ce programme est automatique, bouclé sur lui-même, il montre les possibilités de connections vers imprimante, télévision, lecteur de cassettes, et moniteur vidéo du PHC 25.

LES PROGRAMMES

Le PHC 25 exécute automatiquement tout programme qui lui est désigné. Voici la manière d'introduire un programme :

a) PREPARATIFS :

Avant de faire un nouveau programme, effacer l'ancien pour ne pas risquer de mélanger les deux. Pour effacer, taper NEW suivi d'un RETURN.

b) METHODE :

Le programme est un groupage de lignes commençant par un numéro. Lorsque l'on fait une entrée d'instructions, chaque ligne s'inscrit sur l'écran et il ne faut pas oublier d'appuyer sur la touche RETURN.

c) VERIFICATION DU PROGRAMME :

Taper LIST suivi d'un RETURN . Le programme sera listé sur l'écran dans l'ordre des lignes. On peut arrêter le défilement en appuyant sur la touche ESC. Appuyer à nouveau sur ESC pour continuer. Pour stoppper définitivement le défilement, appuyer simultanément sur CTRL et C.

d) EXECUTION DU PROGRAMME :

Pour exécuter le programme, taper RUN suivi d'un RETURN. L'emploi de la touche ESC permet d'arrêter l'exécution du programme temporairement. Cette commande RUN est programmée sur la touche F1.

e) CORRECTION DE PROGRAMME :

Lorsqu'il y a erreur de message, utilisez le tableau "ANNEXE A". Il y a trois types de corrections :

1) Ajouter des lignes

Il suffit d'inscrire le numéro de la ligne à ajouter, puis de composer le texte de l'instruction désiré. Ne pas oublier de faire RETURN, cette instruction s'installera dans le programme à l'endroit indiqué par le No de ligne.

2) Effacer des lignes

Inscrire uniquement le numéro de la ligne sans aucun texte et appuyer sur RETURN.

3) Correction du contenu d'une ligne

Soit retaper entièrement la ligne en commençant par son numéro, soit l'afficher par la commande LIST suivie du numéro de ligne et corriger les erreurs en utilisant la touche INS/DEL et en déplacant le curseur avec les flèches. Dans les deux cas, il faut faire RETURN après la correction. diffèrents.

f) Conception d'un programme :

Le manuel d'instructions donne toutes les règles de syntaxe du BASIC. Il n'explique pas les significations du BASIC. Les règles sont les suivantes :

1) Entrée des textes inscrits en majuscules exactement comme ils se présentent. Les textes peuvent également être en lettres minuscules. Toutefois, lorsque l'on utilise des guillements (") pour des chaînes de caractères, il faut toujours faire la distinction entre majuscules et minuscules et en particulier pour les noms de fichiers.

2) Dans les explications, les données entre parenthèses ( ) doivent être déclarées.

3) Lorsque deux données ou plus sont séparées par des lignes verticales, cela indique un choix de l'une de ces données.

4) Les données entre parenthèses carrées [ ] sont des options. Lorsqu'elles ne sont pas nécessaires, elles peuvent ne pas être inscrites.

5) Lorsque vous avez trois points (...) cela indique que l'expression immédiatement devant est sujette à répétition.

EMPLOI DU BASIC

1) Forme de la ligne :

Un programme de BASIC consiste en un groupe de lignes. La forme de la ligne est

nnnn INSTRUCTION BASIC
nnnn est le no de la ligne

Les instructions multiples peuvent être faites sur une ligne. Dans ce cas, il faut obligatoirement séparer les instructions par deux points (:)

Le programme basic commence toujours par le numéro de ligne et finit par un RETURN

Le nombre maximum de caractères possibles sur une ligne est de 128.

Les nombres entiers utilisés sont de 0 à 65535.

Les numéros de lignes indiquent l'ordre de stockage et d'excécution chaque ligne de programme en mémoire. Ils facilitent également l'édition, le défilement et l'identification.

2) Les caractères et signes utilisés :

Classification	Caractères
Lettres majuscules	A-Z a - z
Chiffres	0 - 9
Caractères spéciaux	^ + - * / ! $ % & , ()[] : ; : <> . ? = espace

LES CONSTANTES

Il y a deux types de constantes : la chaîne de caractères constante et la valeur constante.

1) LA CHAINE DE CARACTERES CONSTANTE :

Elle est constituée par un ou plusieurs caractères enfermés entre guillemets. La forme générale étant "caractère..."

Exemple : "PHC 25"

NB : Les guillemets (") ne sont pas compris dans la chaîne de caractères.

2) VALEURS CONSTANTES :

Les valeurs constantes sont des nombres positifs et négatifs. Il y a trois types :

a) - NOTATION A POINT (virgule) FIXE :
On obtient un nombre négatif ou positif qui inclut le point décimal avec neuf chiffres significatifs.
Exemple : 33.169

b) - NOTATION A POINT DECIMAL (virgule) FLOTTANT:
On obtient un nombre positif ou négatif exprimé en notation exponentielle. L'exposant va de -38 à +38.
La mantisse peut avoir jusqu'à 9 chiffres significatifs.
Exemple : 123456789 E-38

c) - NOTATION HEXADECIMALE :
La constante hexadécimale est exprimée par une combinaison de caractères de 0 à F précédés par &H
Exemple : &HC000

Les sorties d'inscriptions hexadécimales sont faites en notation décimale.
Les variables sont exprimés en signes (noms variables) qui mémorisent des caractères particuliers dont la valeur peut être changée pendant l'exécution du programme.

La valeur est censée être zéro jusqu'au moment où cette valeur est changée. Bien qu'elle puisse être n'importe quel nombre de caractères (jusqu'à 255), la distinction se fait par les deux premiers caractères. A titre d'exemple :

ABI et ABX

sont considérés comme identiques. Le premier caractère doit obligatoirement être une lettre, après cela vous pouvez avoir des chiffres.
Toutefois, le seul signe accepté est le symbole $

LES VARIABLES

Les variables peuvent être classées en trois catégories :

1) VALEUR VARIABLE :

Utilisée pour mémoriser les valeurs de 9 chiffres significatifs entre

10^{-39} à 10^{+38}

```
Exemple:A,BD,NI,POR,
        A=123456789 E-6 (soit en décimal 123,456789)
        BD=1234 E8      ( "          "    123 400 000 000)
```

2) CHAINE DE CARACTERES VARIABLES :

Utilisée pour mémoriser des chaînes de caractères jusqu'à 255 caractères. Les chaînes de caractères variables utilisent le signe $ après le nom variable.

Exemple : P$,DX$,XI$,CDE$

3) TABLEAU VARIABLE :

Le groupage de data peut être fait sous le terme TABLEAU VARIABLE. On peut identifier le TABLEAU par un caractère entre parenthèses après le nom variable.

Il faut noter que le TABLEAU VARIABLE a le même nombre de caractères entre parenthèses que les dimensions du "tableau".

Exemple : A(3),CN(1),KB(2,4)

N.B Les mêmes chaînes variables que celles utilisées pour les instructions de Basic ne peuvent être utilisées comme nom variable. Une liste figure en ANNEXE C.

****

LES EXPRESSIONS ET OPERATIONS :

L'expression est un groupe de constantes et variables combinées avec les opérateurs pour un calcul de variables , constantes et valeurs. Les opérateurs vont exécuter des programmes d'arithmétique, de logique pour chaque valeur.

Ces opérations sont à classer dans cinq catégories :

a) opérations arithmétiques
b) opérations relationnelles
c) opérations logiques
d) opérations de chaines de caractères
e) fonctions

a) OPERATIONS ARITHMETIQUES :

Opérateurs	Opérations	Exemples
^	puissance	A^B
+	signe positif addition	x+y
-	signe négatif soustraction	x-y
*	multiplication	x*y
/	division	x/y

Exemple:

```
10 LET a=10
20 LET b=4
30 LET x=a^b
40 PRINT x
50 END

RUN
10000
```


b) OPERATIONS RELATIONNELLES :

Utilisées pour la comparaison de deux valeurs. Le résultat est soit juste (-1) ou faux (0). Ce résultat est utilisé pour déterminer le contrôle de défilement du programme.

Opérateurs	Relations	Expressions
=	EGAL	X=Y
<>	DIFFERENT DE	X<>Y
<	PLUS PETIT QUE	X<Y
>	PLUS GRAND QUE	X>Y
<=	PLUS PETIT OU EGAL	X<=Y
>=	PLUS GRAND OU EGAL	X>=Y

c) OPERATIONS LOGIQUES :

Les opérations logiques, comme les opérations relationnelles, donnent des résultats en valeurs justes et valeurs fausses. L'opération logique est effectuée après les opérations arithmétiques et relationnelles.

AND	X	Y	X AND Y
	1	1	1
	1	0	0
	0	1	0
	0	0	0

OR	X	Y	X OR Y
	1	1	1
	1	0	1
	0	1	1
	0	0	0

NOT	X	NOT X
	1	0
	0	1

De même que les opérateurs relationnels déterminent le défilement du programme, les opérateurs logiques ajoutent aussi trois autres relations qui donnent une valeur juste ou fausse.

d) OPERATIONS DE CHAINES DE CARACTERES :

Les opérations de chaînes de caractères sont utilisées pour combiner la chaîne de caractères utilisant le signe +

A titre d'exemple, lorsque

A$="Good";B$="morning"

A$+B$="Good Morning"

A$+B$+"!"="Good Morning !"

e) FONCTIONS :

Le PHC 25 possède des fonctions telles ques ABS et TAN. Egalement des fonctions qui peuvent être définies d'une façon abrégée par l'utilitaire DEFFN.

COMMANDES D'EXPRESSIONS OPERATIONNELLES :

L'ordre de priorité des opérations dépend des facteurs suivants :

a) ordre écrit b) priorité propre à l'opérateur
c) référence aux fonctions d) utilisation des parenthèses

En règle générale, les opérations défilent de gauche à droite dans l'ordre d'écriture, toutefois cet ordre peut changer suivant les facteurs b c ou d. Le degré de priorité de chaque opérateur est visible sur le tableau 2.2. Pour obtenir des références à des fonctions, les opérations fonctions doivent être effectuées en premier lieu. Si l'on utilise des parenthèses, l'opération à l'intérieur des parenthèses est effectuée en premier lieu.

ordre d'exécution	opérateurs
1	signe (positif ou négatif)
2	* / (multiplication ou division)
3	+ - (addition ou soustraction)
4	= , < , > , <> , <= , >=
5	NOT
6	AND
7	OR

MESSAGES D'ERREUR

Un message d'erreur est affiché lorsqu'une erreur est détectée pouvant causer l'interruption du programme. Dans le mode direct cette erreur est affichée comme suit :

XXXXX in nnn

XXXXX est le messsage d'erreur,
nnn est le numéro de ligne de l'erreur.

Le mode direct permet la modification de l'instruction fausse par simple réécriture directement à l'écran en même temps qu'en mémoire :

exemple : syntax error in 100
exemple : illegal function call at 1000
exemple : not enough data

commentaire : la liste complète des messages d'erreur pouvant apparaître est en annexe H.

COMMANDES

1.- CLEAR :

Syntaxe : CLEAR I , [J]

Objet : remettre à zéro toutes les variables numériques et à valeur nulle toutes les valeurs alphanumériques.

La variable I est obligatoire et détermine la taille de la mémoire réservée pour y stocker des données.

La variable J est optionnelle et détermine la limite supérieure de la mémoire utilisable sous Basic.J ne peut pas dépasser la valeur &HF800 (hexa)

exemple : CLEAR 400,&HE050

Commentaires : La taille de mémoire réservée est de 400 caractères et la limite supérieure de la mémoire est fixée à l'adresse E050 (hexa), c'est-à-dire qu'on pourra écrire un programme en langage machine au-delà de l'adresse E050 puisque le programme en basic ne dépassera pas cette valeur.

exemple : CLEAR 200

Commentaires : 500 caractères peuvent être entrés sous forme de données.

****

2.- CLOAD :

Syntaxe : CLOAD"nom"

Objet : charger en mémoire un programme ou un fichier existant sur une cassette.

Exemple : CLOAD"PENDU" (le guillemet après le nom n'est pas obligatoire), puis RETURN

Remarques : CLOAD efface le programme précédent avant de charger celui qui est sur la cassette. Le "nom" est une chaîne dont les six premiers caractères sont significatifs et doivent être identiques au nom sous lequel le programme a été sauvegardé par CSAVE.

S'assurer que le magnétocassette est correctement branché et que la touche PLAY est enfoncée.

NOTE : Si aucun nom n'est précisé, le PHC 25 charge automatiquement le premier programme trouvé sur la cassette. Dans ce cas, il faut taper CLOAD sans guillemet.

Attention aux minuscules et majuscules pour les noms

3.- CLOAD?

Une comparaison est faite entre le programme en mémoire et celui qui est sur la cassette.

L'écran affiche "READY" est affichée s'il est identique

"BAD" dans le cas contraire, il faut alors refaire la procédure de CSAVE à nouveau.

exemple : CLOAD?"TOTO"

****

4.- CONT :

CONT veut dire continuer.

Utilisé pour reprendre l'exécution d'un programme après une suspension par la commande CTRL/C ou après l'exécution d'une instruction STOP.

L'exécution reprend à l'endroit où elle a été interrompue.

CONT n'est plus utilisable si une modification quelconque a été faite dans le programme, ni après une instruction END. L'écran affiche "Can't continue"

****

5.- CSAVE :

Syntaxe : CSAVE"TOTO"

Objet: Sauvegarde sur cassette d'un programme en mémoire.

Remarques : Chaque programme sauvegardé est identifié par un nom.

Le BASIC copie le programme existant en mémoire sur la cassette lorsque la commande CSAVE est exécutée.

Elle se sert des six premiers caractères pour le "nom"

Avant l'utilisation d'un CSAVE, il est nécessaire de vérifier que le magnétophone est branché correctement et que les touches RECORD et PLAY sont bien enfoncées.

****

6.- EXEC :

Syntaxe : EXEC&HXXXX

XXXX est l'adresse hexadécimale ou ira le programme.

objet : Le programme en basic ira à l'adresse indiquée par la variable XXXX pour éxécuter un programme écrit en langage machine (hexadécimal). Le programme aura été écrit grâce à l'instruction POKE et devra se finir automatiquement par la valeur (&HC9) (RET) qui permettra un retour au langage Basic.

exemple :
```
 10 CLEAR 100,&HF000
 20 LET J=00
 30 INPUT"valeur Hex";A$
 40 N =VAL("&H"+A$)
 50 IF A$="FF" THEN 90
 60 POKE&HF000+(J),N
 70 J=J+1
 80 GOTO 30
 90 END
100 CLS
110 EXEC&HF000
120 PRINT"SORTIE"
130 END
```

```
RUN
valeur Hex? 3E |
valeur Hex? 41 | Mettre A dans l'accumulateur du Z80

valeur Hex? 32 |
valeur Hex? 8F | Mettre A à l'adresse 608F
valeur Hex? 60 | (5ème ligne 16ème colonne)

valeur Hex? C9  RET

valeur Hex? FF  Sortie de la routine 10-90
Ready
```



```
RUN100
```

La lettre A sera écrite en 5eme ligne 16eme colonne. Dans cet exemple les instructions hexadécimales ont été écrites par une saisie manuelle au clavier. Il est aussi possible de les stocker par l'instruction DATA.

exemple :
```
 10 CLEAR 100,&HF000
 20 J=-1
 30 FOR K=1TO6
 40 READ A$
 50 J=J+1
 60 POKE&HF000+(J),VAL("&H"+A$)
 70 NEXT
 80 DATA 3E,45,32,8F,60,C9
 90 CLS
100 EXEC&HF000
110 PRINT"SORTIE"
120 END
```

Dans cet exemple on voit que les datas servent d'instruction pour la commande EXEC de la ligne 100. Ne pas oublier de finir les datas par C9.

****

7.- LIST :

Syntaxe : LIST nn-mm

Objet : Cette commande affiche sur l'écran une liste du programme stocké en mémoire.

Si les numéros de lignes sont absents, le programme entier est listé.

exemple : LIST

Si les numéros de lignes de début et de fin sont donnés, toutes les lignes entre ces deux valeurs seront listées

exemple : LIST10-100

Si l'on désigne le numéro de la ligne de début suivi du signe "-", le listing commencera à cette ligne et continuera jusqu'à la fin.

exemple : LIST10-

Si l'on commence par le signe"-" suivi d'un numéro de ligne finale, le listing commencera au début du programme et se terminera à la ligne indiquée.

exemple : LIST-100

Si l'on indique un numéro ,cette ligne seule sera affichée.

exemple : LIST10

****

8.- LLIST

Lister sur l'imprimante tout ou partie du programme existant en mémoire.

Remarques : La commande LLIST assume l'utilisation d'une imprimante 80 caractères. La méthode d'opération est identique à la commande LIST.

9.- NEW

Syntaxe : NEW (Return)

Objet : Effacer le programme présent en mémoire et remettre à zéro toutes les variables.

Remarques : La commande NEW doit être utilisée avant d'entrer un nouveau programme.

****

10.- RUN

Syntaxe : RUN (Return)

Objet : Lancement de l'exécution du programme stocké en mémoire.

Si le numéro de ligne est indiqué, c'est à cette ligne que l'on débute, sinon c'est au début.

exemple : RUN100
RUN
F1

La touche F1 simule cette commande.

INSTRUCTIONS IMMEDIATES

Lorsque des inscriptions sont inscrites directement, c'est-à-dire sans numéro de ligne, on les appelle instructions immédiates.

Dans ce cas, il y a exécution aussitôt qu'elles sont entrées.

Les instructions immédiates peuvent également être multiples. (voir chapitre suivant).

Exemple : FOR I=1 TO 6 : PRINT I : NEXT I

1 2 3 4 5 6

INSTRUCTIONS DU BASIC

La combinaison des diverses instructions en BASIC avec les numéros de lignes et le groupage de ces lignes, forment le programme.

1.- CLS

Tous les caractères affichés sur l'écran sont effacés.

****

2.- COLOR

Syntaxe : color a,b,c

Objet : Désignation de la couleur des caractères ou graphiques affichés sur l'écran.

Même si on désigne les mêmes chiffres, la couleur affichée est différente selon le mode écran choisi. 1,2,3,4

Mode caracteres (screen 1)

Dans ce mode, la couleur de l'écran est désigné par les valeurs a et c sans tenir compte de b.

La couleur de l'écran entier est changée en exécutant l'instruction COLOR et ensuite CLS.

Il n'y a pas de graphique possible en mode 1.

a	b	c	Caractères
1		1	BLANC sur fond VERT
1		2	BLANC sur font ORANGE
2		1	NOIR sur fond VERT
2		2	NOIR sur fond ORANGE

Exemple :

```
10    SCREEN 1, 1, 1
20    COLOR 1,, 2
30    CLS
40    PRINT "ABCD"
50    END
RUN
```

REMARQUE : Dans cet exemple COLOR 1,2 est identique à COLOR 1,1,2
ou encore à COLOR 1,2,2

****

Mode graphique 64x48 (SCREEN 2) :

Dans ce mode graphique, l'instruction COLOR est valable seulement pour les caractères et graphiques affichés sur l'écran après l'exécution de l'instruction. La couleur du graphique est déterminée par a et c. Neuf (9) couleurs peuvent être utilisées, mais seulement 4 couleurs pour les caractères.
La valeur de b affecte la couleur du fond d'écran.

a	b	c	caractère	fond	graphiques
0	0	1	vert	noir	noir
1	1	1	vert	vert	vert
2	2	1	vert (inversé)	jaune	jaune
3	3	1	orange	bleu	bleu
4	4	1	orange (inversé)	rouge	rouge
5	5	1	orange (inversé)	blanc	blanc
6	6	1	orange (inversé)	bleu clair	bleu clair
7	7	1	orange (inversé)	violet	violet
8	8	1	orange (inversé)	orange	orange

a	b	c	caractère	fond	graphiques
0	0	2	orange	noir	noir
1	1	2	orange	blanc	blanc
2	2	2	orange (inversé)	bleu clair	bleu clair
3	3	2	vert	violet	violet
4	4	2	vert (inversé)	orange	orange
5	5	2	vert (inversé)	vert	vert
6	6	2	vert (inversé)	jaune	jaune
7	7	2	vert (inversé)	bleu	bleu
8	8	2	vert (inversé)	rouge	rouge

Exemple :

```
10 SCREEN 2, 1, 1
20 COLOR 2, 5, 1
30 CLS
40 PRINT "a"
RUN
```

Dans cet exemple les caractères seront noir sur fond vert clair.
Les graphiques seront vert clair sur fond blanc sachant que:

$1 \leq a \leq 8$, $1 \leq b \leq 8$, $1 \leq c \leq 2$. Il existe 128 possibilités différentes.

****

MODE GRAPHIQUE 128 x 192 (SCREEN 3)

Dans ce mode graphique, l'instruction COLOR n'est valable que pour les caractères et graphiques affichés sur l'écran après l'exécution de ladite instruction.

a	b	c	caractère ou graphiques
1		1	vert
2		1	jaune
3		1	bleu
4		1	rouge
1		2	blanc
2		2	bleu clair
3		2	violet
4		2	orange

EXEMPLE :

```
10 SCREEN 3, 1, 1
20 COLOR 2, 1, 1
30 CLS
RUN
```

La couleur de fond est déterminé par les valeurs de b et c comme dans le mode SCREEN 2, mais b est limité entre 0 et 4, ce qui procure 64 possibilités differentes.

****

GRAPHIQUE MODE 256 x 192 (SCREEN 4)

Dans ce mode graphique, l'instruction COLOR n'est valable que pour les caractères et graphiques affichés sur l'écran après l'exécution de ladite instruction.

a	b	c	caractères ou graphiques	FOND
1	0	1	VERT	NOIR
0	1	1	NOIR	VERT
1	0	2	BLANC	NOIR
0	1	2	NOIR	BLANC

NB. Dans le cas où l'on désire un changement de couleur de l'écran entier, faire la couleur b identique à a puis exécuter l'instruction COLOR et CLS

3.- CONSOLE (a),(b)

Objet : Désigne les limites de déplacement vers le bas et le haut de l'écran.

a : ligne de depart du scroll

b : longueur de la ligne du scroll

exemple :
```
10 CONSOLE 0,15
20 PRINT "TITRE"
30 CONSOLE 1,15
40 CLS
50 CONSOLE 3,12
60 PRINT "DEBUT"
```

Commentaire :
en 10 on peut utiliser les 16 lignes de l'écran
en 20 on écrit TITRE sur la première ligne
en 30 on ne peut utiliser que les 15 dernières lignes
en 40 on efface les lignes définies en 30
en 50 on réduit la zone de travail de la 3ème à la 15ème ligne.

****

4.- CTOFF :

Objet : Fermer l'interrupteur de commande à distance du magnétophone (REMote Control) arrêtant ainsi le défilement de la cassette.

****

5.- CTON :

Objet : Mettre le magnétocassette en marche dans le cas ou la touche PLAY (...) enfoncée.

La commande à distance du magnétocassette est enclenchée par cette commande, démarrant ainsi la cassette.

Vérifier que la touche PLAY du magnétophone est bien enclenchée.

6.- DATA ("constant" "constant"...)

Objet : Utilisé pour enregistrer dans un programme des valeurs numériques et des caractères constants qui seront appelés par des instructions READ.

Les instructions DATA peuvent être inscrites à n'importe quelle place du programme. On peut mettre autant de constantes que la longueur de la ligne le permet à l'intérieur d'une instruction DATA.

L'instruction READ lira les instructions DATA dans l'ordre des numéros de lignes. Les données DATA peuvent être stockées dans ces instructions et considérées comme une liste continue de DATA.

DATA exp,exp,etc...

Cette instruction permet de conserver des données dans un programme sous forme de liste. Le programme peut accéder séquentiellement aux données en commençant par la première à l'aide de l'instruction READ. Les chaines de caractères doivent être encadrées de guillements. Les lignes DATA peuvent se situer n'importe où dans le programme.

exemple :
```
10 FOR I=1 TO 3:READA$(I):NEXT I
20 FOR I=1 TO 3:READB(I):NEXT I
30 FOR I=1 TO 3:PRINT A$(I);"=";B(I);"F":NEXT I

100 DATA"PAIN","VIN","JAMBON",2,6.50,10

RUN

PAIN=2F
VIN=6.50F
JAMBON=10F
```

COMMENTAIRES :

10 chargement du tableau A$ avec les 3 premières données
20 chargement du tableau B avec les 3 données suivantes
30 affichage des tableaux
100 liste des données

Nota : Si en cours de programme il est nécessaire de relire les lignes DATA,il faut réinitialiser le pointeur de DATA à la ligne désirée par l'instruction RESTORE.

****

7.- DEFFN "nom" "Liste des paramètres"

Objet : Définir une fonction écrite par l'utilisateur et lui donner un nom.

- Le "nom" doit être un nom de variable ou le nom d'un caractère variable. Le nom précédé de FN + "nom" devient celui de la fonction.

- La "liste des paramètres" comprend les variables qui seront échangées contre des valeurs réelles lors de l'appel de la fonction. Les articles de la liste sont séparés par des virgules.

- La "définition fonction" est une expression qui indique l'opération de la fonction. Elle est limitée à une ligne.

- La définition de la fonction avec l'instruction DEFFN doit être faite avant d'être utilisée.

Un message d'erreur apparaît si la fonction est utilisée avant d'être définie.

- DEFFN ne peut être utilisé en mode direct.

Exemple : 10 DEFFNA(X)=2*X+X/3

20 PRINT FNA(3)

RUN
Le résultat sera 7.

****

8.- DIM :

DIM "nom variable" "valeur maximum souscrite"

"liste de variables indicées"

Objet : Préciser les valeurs maximales des indices pour les tableaux et réserver la place utile en mémoire. En même temps on assure la place utile pour ranger la mémoire.

Lorsqu'un tableau de variables est utilisé sans instruction DIM, la valeur maximum est considérée comme étant 10 lignes en 3 dimensions.

Une erreur apparaît si l'on fait appel à une valeur plus grande.

La valeur minimale est zéro.

Exemple : DIMB$(20,10),C(30),E(10,3,5,7)

B$ Tableau de caractères à 20 lignes et 10 colonnes
C Tableau numérique de 30 lignes
E Tableau numérique de 4 dimensions

****

9.- END :

Objet : Terminer l'exécution d'un programme.

L'instruction END peut être placée n'importe où dans le programme pour terminer l'exécution.

Le message BREAK n'est pas imprimé lorsque l'instruction END est exécutée.

La commande CONT ne peut être utilisée après.

****

10.- FOR - NEXT

FOR variable = X TO Y (STEP Z)

NEXT variable,variable ... (X, Y, Z sont

des expressions numériques).

Remarques : FOR - NEXT répète et exécute une chaîne de commandes le nombre de fois désigné.

La première expression X, est la valeur initiale, Y est une valeur limite à ne pas dépasser.

Les instructions entre FOR et NEXT sont exécutées autant de fois que nécessaire avec chaque fois incrémentation du compteur variable de la quantité indiquée après l'instrution STEP pour que le compteur dépasse la limite Y. Dans un tel cas, l'exécution passe à l'instruction suivante NEXT.

Si (X - Y) X est négatif, le corps de la boucle n'est pas exécuté, c'est-à-dire si Y est inférieur à X avec un pas positif, ou si Y est supérieur à X avec un pas négatif.

Deux boucles FOR-NEXT peuvent-être, soit l'une à l'intérieur de l'autre, soit complètement séparées.

On appelle cela des boucles imbriquées.

Il faut également éviter de sauter au milieu d'une boucle FOR-NEXT à partir d'un point extérieur.

La variable de contrôle prend seulement une valeur correcte lorsque son instruction FOR est exécutée.

La seule limitation de la boucle imbriquée est une limitation de mémoire.

Le contrôle du compteur de la boucle intérieure doit être indépendant et venir avant celui de la boucle extérieure.

Exemple :

```
 10 FOR I=1 TO 10 STEP 2
 20 FOR J=2 TO 5 STEP 1
 30 FOR K=0 TO 20
100 NEXT K
110 NEXT J
120 NEXT I
130 END
```

****

11.- GOSUB - RETURN

Syntaxe : GOSUB adresse

Objet : Aller à un sous-programme et revenir.

Remarques : Le défilement indiqué par le numéro de ligne est la première ligne du sous-programme. On peut appeler un sous-programme autant de fois que l'on veut dans le programme. Un sous-programme peut en appeler un autre. La seule limitation étant la dimension de mémoire.

Le sous-programme est terminé par une instruction RETURN qui renvoie à l'instruction qui suit celle qui a appelé le sous-programme.

exemple :

```
 10 PRINT "BCD"
 20 GOSUB 100
 30 PRINT "DEF"
 40 END
100 PRINT "123"
110 RETURN
```

****

12.- GO TO - GOTO numéro de ligne

Saut inconditionnel à une ligne qui a été désignée.

Remarques : Une erreur est signalée si la ligne désignée n'existe pas.

Si la ligne désignée contient une instruction, cette instruction est exécutée, puis celles qui suivent,

Exemple :

```
10  A = 1
20  PRINT A
30  A = A+1
35  IF A=10 THEN STOP
40  GOTO 20
```

13.- IF - THEN - ELSE

IF expression THEN instruction numéro de ligne ELSE "instruction" "numéro de ligne".

Suivant les résultats d'un test, l'on prendra une décision ou l'on choisira une suite pour l'exécution du programme.

Si le résultat du test est vrai, c'est-à-dire différent de zéro, alors l'instruction THEN suivra jusqu'à ELSE ou la fin de la ligne et sera exécutée.

Si l'expression est égale à zéro (valeur fausse), seules les instructions suivant ELSE, si elles existent, seront exécutées.

Les instructions multiples IF - THEN - ELSE peuvent être écrites en une seule ligne.

exemple :

```
  5 A=-5 : B=-10
 10 IF A< >0 THEN B=10 ELSE B=0:GOTO100
 30 PRINT A,B
 40 END
100 PRINT A,B
110 A=A+1
120 GOTO 10
```

```
RUN -5   0
    -4   0
    -3   0
    -2   0
    -1   0
     0  10
```

14.- INPUT

INPUT "chaîne de caractères", liste de variables .

Entrée de données par le clavier pendant l'exécution d'un programme.

Remarques : Lorsqu'une instruction INPUT est exécutée, le point d'interrogation est affiché, indiquant une demande de l'entrée des données. S'il y a une " chaîne de caractères", elle est affichée devant le point d'interrogation (?).

En appuyant sur la touche RETURN, les données entrées sont affectées aux variables dans l'ordre de la liste. Les différentes données multiples sont séparées par une virgule.

Exemple :

```
10    INPUT  A,B
RUN
?
10    INPUT "A=" ; A
RUN
A=?
```


15.- INPUT #

INPUT# numéro de fichier, liste de variables

Lecture de données dans un fichier séquentiel et affectation des valeurs aux variables de la liste.

Remarques : Le numéro du fichier est le numéro associé au fichier au moment de son ouverture.

La liste contient le nom des variables où l'on doit affecter les données existantes sur le fichier.

Il faut que les types concordent et il n'y a pas de point d'interrogation sur l'écran comme dans le cas "INPUT".

Exemple :

INPUT# - 0 ... clavier

INPUT# - 1 ... magnétophone

Lorsque l'on entre des données par l'intermédiaire du magnétophone, il est nécessaire d'enregistrer ces données par une instruction PRINT£

Exemple :

```
10 INPUT#- 1, A$

20 PRINT  LEN(A$)
```

****

16.- KEY

Syntaxe : KEYn,"Texte"

Les touches F1 F2 F3 F4 sont reprogrammables en mode normal et en mode SHIFT. Ce qui donne 8 touches au total.

Une touche peut contenir 8 caractères exécutables au maximum.

exemple : KEY1,"CONSOLE" la touche F1 écrira console

KEY7,"CTON"+CHR$(13) la touche F3 mettra le moteur du magnétocassette en marche.

17.- LCOPY

Envoi de l'information contenue sur l'écran vers l'imprimante.Aussi appelée "HARD COPY". Ne marche que sur imprimante graphique.

****

18.- LET

LET nom variable expression .

La valeur de "expression" est substituée et devient une variable.

En fait, le signe égal (=) suffit pour constituer l'équation variable. L'instruction LET a le même but.

Exemple :

```
10      LET A = B + 1

20      LET C = 3

110     A = B + 1

120     C = 3
```

Les lignes 10 et 110 ou 20 et 120 sont identiques.

****

19.- LINE

Forme LINE(X1,Y1)-(X2,Y2),(couleur),(BF)

Objet : Relier les points par une ligne ou les insérer dans un carré.

Deux points désignés par des références données sont reliés par une ligne qui a une couleur désignée.

Lorsque "B" est désigné, un rectangle est dessiné utilisant les deux points comme angles opposés.

De plus, si F est désigné, le rectangle est coloré.

Exemple : LINE(5,5)-(20,20),3

Les points de coordonnées 5,5 et 20,20 sont reliés par une ligne de la couleur 3

Exemple : LINE(5,5)-(20,20),3,B

Les points 5,5 et 20,20 appartiennent à un carré. Ils sont les deux points opposés.

Exemple : LINE(10,8)-(190,130),2,BF

Un carré colorié de couleur 2 passe par les points de coordonnées 10-8 et 190-130

Remarque : $1 \leq b \leq 4$

Les définitions en ligne et en colonne sont assujetties à la définition du SCREEN 1,2,3,4

20.- LOCATE

Syntaxe : locate,h,v
"position horizontale", "position verticale"

Le curseur est bougé vers une position optionnelle sur l'écran qui est exprimée par une position horizontale et verticale.

Le coin gauche en haut de l'écran est 0,0.

La "position horizontale" est de 0 à 31 pour le mode 1,2,4 d'écran

La "position verticale" est de 0 à 15 pour tous les modes d'écran

Toutefois, lorsque l'écran 3 est utilisé, la position horizontale s'arrête à 15.

exemple :
```
10 FOR I=1 TO 100
20 LOCATE 10,10
30 PRINT I
40 NEXT
RUN
```

****

21.- LPRINT expression

Ecriture de données sur l'imprimante.

Cette instruction est identique à PRINT sauf que la sortie se fait sur l'imprimante connectée au PHC 25

exemple : LPRINT "BONJOUR"

****

22.- ON - GOSUB

ON - GOTO

ON condition GOTO ligne numéro 1 , ligne numero 2,etc
ON condition GOSUB ligne numéro 1, ligne numéro 2,etc

Objet : branchement multiple suivant la valeur de l'expression. C'est cette valeur qui détermine sur quelle ligne le programme sera branché. Si la valeur est I, le programme sera branché sur I. Si l'expression n'est pas intégrale (entière) les décimales seront arrondies. Si la valeur de l'expression est zéro ou négative ou si elle est plus grande que les numéros de ligne, le programme passera à la ligne suivante.

Exemple :
```
10 K=0
20 ON K+1 GOTO 100,200,300,400
```

Commentaires : Suivant la valeur initiale de K (0,1,2,3), le saut sera fait à l'adresse 100,200,300,400. Ici K=0, le saut est à 100. Si en 10 on met K=2 le saut sera en 300, puisque K+1=3.

23 .-OUT

forme OUT I,J

Objet: Cette instruction permet d'envoyer sur le port I l'octet J

```
exemple : OUT 32,100
          OUT &H20,&H64
```

Les deux exemples sont identiques. Les valeurs I et J peuvent être exprimées en décimal ou en hexadécimal. Dans ces exemples la valeur 64H (HEXA) sera envoyée sur le port 20H (HEXA). Cette instruction ne peut être utile que pour les informaticiens confirmés car elle émule du hardware.

****

24.- PAINT

PAINT (X,Y),color,périmètre

Syntaxe : PAINT (x,y),c,p

Les coordonnées x et y définissent 1 point se situant dans une zone de l'écran. Cette zone est peinte avec la couleur C sauf le périmètre de couleur P.

```
exemple : 10 line(0,0)-(255,0),2,BF
          11 line(255,0)-(255,191),2,BF
          12 line(255,191)-(5,186),2,BF
          13 line(5,186)-(0,0,),2,BF

          20 Paint (80,80),4,2

          30 goto 30
RUN
```

Commentaires : un cadre de couleur 2 (vert) est dessiné autour de l'écran, l'instruction en 20 peint la zone d'écran auquel appartient le point 80,80 de la couleur 4 (orange) sauf la zone de couleur 2 (vert).

1 = vert foncé 2 = vert clair 3 = rouge 4 = orange	SCREEN 1,3,4 in color 1,1,1	1 = vert clair 2 = jaune 3 = bleu foncé 4 = rouge	SCREEN 2 in color 1,1,1

Ces couleurs sont valables pour les fonctions Paint, line, etc...qui permettent le tracé de graphiques couleur.

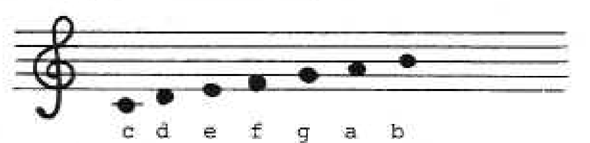
25.-PLAY "Chaîne de caracteres A"
"Chaîne de caractères B"
"Chaîne de caractères C"

Syntax:PLAY"ox[sx] [mx] [vx] [lx] [tx] [rx] [x+] [x-] (nx§abcdefg"

Objet : la musique est produite par le générateur de son (vendu en option).

Les lettres A à G indiquent les sons suivants :

Les fonctions entre crochet sont facultatives.



Dans l'instruction "PLAY" à chaîne de caractères, les caractères suivants sont utilisés pour donner des significations spéciales. Voir ci-dessous :

sX	désigne la forme de l'enveloppe (se référer à l'annexe D.) ($0 \leq X \leq 15$)
mX	désigne la période de l'enveloppe ($1 \leq X \leq 65535$)
vX	désigne le volume ($0 \leq X \leq 15$) initial 8
lX	désigne la durée du son ($0 \leq X \leq 64$)
oX	désigne l'octave (l'octave plus haute est 04) initial 4 ($1 \leq X \leq 8$)
tX	désigne la vitesse (tempo) du son initial 170 ($32 \leq X \leq 255$)
rX	désigne la durée de la période de silence (aucun son) ($1 \leq X \leq 64$)
x+	le son est remonté d'un demi-ton
x-	le son est descendu d'un demi-ton
nX	désigne le pitch (ou niveau de hauteur) du son.

exemple : PLAY"o4 l8 v2 s5 a"
- a : note la
- s5 : enveloppe 5 voir annexe D
- v2 : volume 2
- l8 : longueur 8
- o4 : octave 4

```
exemple 1 : 10 PLAY"o6l32c"
            20 GOTO 10

exemple 2 : 10 PLAY"o4lcdfgabo5c"
            20 goto 10
```

Il est important d'utiliser les minuscules.

exemple 3 :

```
10 PLAY "o7fardl34fafdbcfao5farecferafbcfl56gbfddaeedaeaea
ddcfffdfbfebacdfedaeo4l29ggabgagafdfdfaffaddeeadaddbccffcf
do7ffacca
```

26.- POKE Expression 1, expression 2

PEEK Expression 1

Objet : Ranger ou charger un octet à une adresse désignée de la mémoire.

L'expression 1 est l'adresse de l'octet

L'expression 2 est la valeur qui doit y être placée.

L'expression 2 doit être contenue entre 0 et 255.

La fonction qui inverse l'opération de POKE est PEEK.

Les instructions POKE et PEEK servent couramment pour une bonne utilisation de la mémoire, pour charger des sous-programmes, pour transférer des paramètres et résultats en langage machine.

Exemple :
```
POKE &H6010,&H81
POKE 4053,24
PRINT PEEK(16388)
PRINT PEEK(&HC000)
```

Voir exemple d'application sur l'instruction EXEC.

****

27.- PRESET

Objet : Effacer le point de coordonnées (X,Y) sur l'écran.

Exemple :

```
 5 SCREEN 4,1,1TO 20
 6 CLS
 9 FOR X=1TO100
10 PSET (X,100),3
11 IF X>20 THEN PRESET (X,100)
12 IF X>50 THEN PSET(X,100),3
15 NEXT
20 GOTO 20
```


28.- PRINT

Ecrire des données sur l'écran.

Remarques : si la liste d'expressions est absente, une ligne blanche s'en suivra. La position de chaque élément de la ligne est déterminée par la ponctuation.
Dans le cas d'expression de caractères, les caractères doivent obligatoirement être encadrés entre guillemets.
Lorsque l'on écrit un nombre multiple d'expressions, chacune doit être séparée par une virgule (,) ou point-virgule (;).
Chaqe ligne est divisée en zones de 16 caractères de long. Une virgule entre deux expressions de la liste fait que l'expression suivant la virgule est imprimée au début de la zone suivante.
Si c'est un point-virgule, l'expression se fait immédiatement après la dernière expression écrite.
Un ou plusieurs espaces entre deux expressions ont le même effet que le point virgule.

Si la ligne est plus longue que la largeur de l'écran, elle est automatiquement continuée sur la ligne suivante. Lorsque la ligne se termine par une virgule ou un point-virgule, la prochaine instruction PRINT continuera la ligne commencée.

```
10      A$="PROGRAMME"
20      B$="No.":C=15
30      D$="exemple de programme"
40      PRINT D$
50      PRINT
60      PRINT A$;B$;C
70      PRINT "suivant"
RUN
exemple de programme
PROGRAMME       No. 15
suivant
```

****

29.- PRINT #

Objet : Ecriture des données dans un appareil désigné. La désignation de l'appareil se fait par le numéro suivant PRINT #

Exemple :

PRINT# - 0, "LUNDI"
"LUNDI" apparait sur l'écran
PRINT# - 1, AB
AB est enregistré sur la cassette du magnétophone
PRINT# - 3, A$
C'est l"imprimante qui a été sélectionnée pour imprimer la chaine de caractères A$.

****

30.- PSET(X,Y),c

Objet : Un point ayant une couleur désignée est affiché à l'endroit spécifié par les coordonnées X,Y

Exemple : PSET(7,12),2
La couleur 2 est affichée au point (7, 12) sur l'écran.

La couleur c dépend du screen 1,2,3,4 et du color choisi. Il y a 256 possibilités en tout.

****

31.- READ liste des variables

Objet : lecture des valeurs dans une instruction DATA et affectation aux variables citées.
Une instruction READ doit toujours être utilisée en relation avec une ou plusieurs instructions DATA.
L'instruction READ distribue les valeurs données dans l'instruction DATA aux variables mentionnées.
Ces variables sont numériques ou de type chaîne et doivent s'accorder, sinon il peut y avoir "erreur de syntaxe".
S'il reste des DATA inutilisés, le READ suivant les utilise.
S'il n'en reste pas, ils seront ignorés.
On peut relire les DATA grâce à l'instruction RESTORE.

Exemple :

```
read A,B,C$     à A est affecté  10
                à B  est  affecté 20.3
restore         à C$ est affecté bonjour

read d,e,f$     à d est affecté  10
                à e est affecté  20.3
                à f$ est affecté bonjour

DATA 10,20.3, bonjour
```

****

32.- REM

Objet : Insérer des remarques et des notes dans le programme.
Bien que l'instruction REM ne soit pas exécutable, elle figure dans la liste avec le contenu du texte qui l'accompagne.
L'instruction REM ne peut pas être continuée en la séparant des autres instructions par un point-virgule (;)

33.- RESTORE

Objet : Permettre de relire à partir d'une certaine ligne.

Après l'instruction RESTORE, le premier READ prend ses valeurs dans la première instruction DATA.

Lorsque le numéro de ligne est spécifié, la lecture se fera à partir de cette ligne.

Si la ligne désignée n'existe pas, l'écran affichera :

"Undefined line number"

****

34.- SCREEN

Syntaxe : SCREEN a,b,c

Le PHC 25 a deux pages d'écran qui sont définies à la mise sous tension.

mode 1 page d'écran mémoire disponible 14265 caractères

mode 2 pages d'écran mémoire disposnible 8121 caractères

Il ne faut pas confondre le choix du nombre de pages utilisées et la façon d'utiliser ces pages.

Evitez d'utiliser des graphiques sur page 1 car, bien que l'on puisse les afficher, ils seront détruits.

En cours de programme, si vous ne désirez pas éteindre le PHC 25 et changer de mode page écran (ce qui correspond à la question posée à l'allumage "SCREEN ?") il faut appliquer le programme suivant

2 pages vers 1 page	1 page vers 2 pages
poke&HFB58,247	clear50,&HE000
poke&HFB56,1	poke&HFB58,223
clear50,&HF800	poke&HFB56,20

Il y a 4 possibilités d'utilisation d'une page suivant le tableau ci-dessous
C'est la valeur a qui fixe le choix.

a	1	2	3	4
texte	16L 32C	16L 32C	16L 16C	16L 32C
définition graphique	0	64x48	128x192	256x192

La valeur b désigne le numéro de la page affectée par les instructions qui suivent. La valeur c désigne le numéro de la page affichée à l'écran.

exemple : SCREEN 4,2,2

35.- SOUND

Forme : Sound "register number" "data"

Objet : Une sortie de son est produite directement en envoyant les commandes au générateur de son. La sortie du son désiré peut être faite en désignant l'information (DATA) qui est dans le registre et les registres de 0 à 13 qui sont listés ci-dessous

Registre	Fonctions du registre	Gamme
0	Désigne le niveau de son	0 255
1	du canal A	0 15
2	désigne le niveau de son	0 255
3	du canal B	0 15
4	désigne le niveau de son	0 255
5	du canal C	0 15
6	désigne la fréquence du bruit	0 31
7	" " bruit et canal	0 63
8	désigne le volume du canal A	0 16
9	" " " B	0 16
10	" " " C	0 16
11	désigne la période et comment	0 255
12	le son est produit	0 255
13	désigne la forme du son voir annexe E	0 15

Désignation du "pitch" ou niveau du son.

Pour produire un son de X Hz en canal A, voici la formule :

P = 1.99675 * 10.6/16/X

R1 = INT (P/256)

RO = INT (P-R1 * 256)

La sortie d'un son de X Hz est faite en désignant le registre 0.1 à la valeur exprimée par RO, R1.

****

Désignation de la fréquence du bruit.

Lorsque l'on désire une fréquence de bruit de X et que la valeur placée en registre 6 est Y, nous obtenons :

Y = INT (1.99675 X 106/16/X)

Désignation du canal et du bruit :

Désignez le canal par lequel le son sera produit et si l'on doit mélanger le bruit.

Pour produire	une sortie	de son	du canal	A		-1
"	"	"	"	B		-2
"	"	"	"	C		-4
"	"	bruit	"	A		-8
"	"	"	"	B		-16
"	"	"	"	C		-32

Trouvez la valeur de l'article approprié dans le tableau ci-dessus et ajouter 63 à cette valeur.

Exemple : Lorsque vous faites un son du canal B :

Désignez 61 (63-2) au registre 7

Lorsque vous faites des sons sur les canaux A,B,C et le mélange de sons du canal B,

Désignez 40 (63-1-2-4-16)

- désignation du volume du son et production du son.

Le volume peut être désigné à une valeur entre 0 et 15.

0 étant le silence
15 le volume maximum

Lorsque 16 est désigné, cela indique le choix par lequel le son est produit. C'est au niveau du registre 13 que ce choix se fait.

- désignation de la période pour la production du son :

Lorsque cette période sélectionnée sur le registre 13 est X et les valeurs désignées sur les registres 11 et 12 sont Y, Z sera : P = 1.99675*10.66/256*X

Z = INT (P/256)

Y = INT (P - Z * 256)

La forme de la production du son correspond à PLAY ou S.

(Se référer à l'annexe E pour ces valeurs).

****

36.- STOP

Arrête le programme en exécution.

Le STOP peut être mis à n'importe quel endroit du programme pour arrêter son exécution.

Toutefois, contrairement à l'exécution END, lorsque STOP est exécuté, le message apparaît :

BREAK in nnnnn

exemple : STOP

****

37.- SLOAD

L'information de l'écran mémorisée par la cassette est enregistrée dans la méloire de l'écran.

Si la page de l'écran utilisée n'est pas celle qui a été mémorisée, la commande SLOAD est arrêtée et une ERROR est affichée.

exemple : SLOAD"TOTO"
Attention aux majuscules et aux minuscules.

****

38.- SSAVE

L'information affichée sur l'écran est sauvegardée sur la cassette du magnétophone.

SSAVE peut servir de commande ou d'instruction.

Aavant l'exécution d'un SAVE, s'assurer que le magnétophone est correctement branché et que les touches RECORD et PLAY sont bien enfoncées.

exemple : SSAVE"TOTO"

****

FONCTIONS INCORPOREES :

Les chiffres significatifs des valeurs fonctions COS, EXP, SIN, TAN, et LOG sont de 6 chiffres. Les chiffres au-delà peuvent contenir des erreurs.

39.- FONCTION ABS

Obtient la valeur absolue de l'argument.

```
Exemple : PRINT ABS(9*(-2))
          18
```

****

40.- FONCTION ASC

Obtient le code ASCII du premier caractère d'une chaîne de caractères X$.

```
Exemple : 10 A$="TEST"
          20 PRINT ASC(A$)
          RUN
          84 (parce que T=84)
```

NB. Pour l'opération inverse c.à d. pour aller du code ASCII à un affichage de caractères, se référer au paragraphe traitant le code CHR$

****

41.- FONCTION CHR$

Syntaxe : CHR$(x)

Objet : générer un caractère de code ASCII égal à la valeur X. Cette fonction est utilisée pour envoyer un caractère spécial.

Exemple : PRINT CHR$(42)

Exemple : PRINT CHR$(20)+CHR$(49)

****

42.- FONCTION COS

On obtient le cosinus de X exprimé en radians. Le résultat est donné en simple précision.

```
Exemple : PRINT COS(0.4)
          .921060995
```

43.- FONCTION CSRLIN

Obtient la position de l'axe Y à partir de la position du curseur.

```
Exemple : 10 LOCATE 12,7
          20 PRINT CSRLIN
          RUN
          7
```

****

44.- FONCTION EXP

Obtient la valeur de e à la puissance X

```
Exemple : PRINT EXP(6)
          403.428794
```

****

45.- FONCTION FRE

Obtient le montant de la mémoire disponible.

Exemple : PRINT FRE(X)

Affiche le nombre d'octets restant dans la mémoire que le Basic n'utilise pas.

PRINT FRE(X$)

Affiche le nombre d'octets restant dans la zone de caractères.

****

46.- FONCTION INKEY$

Ce caractère est obtenu lorsque l'on appuie sur le clavier. la chaîne nulle est obtenue lorqu'on n'appuie pas sur le clavier.

Exemple : 130 IF INKEY$=""THEN GOTO 130

Ceci indique qu'il faut attendre la ligne 130 avant d'appuyer sur une touche.

****

47. FONCTION INP

Obtient la lecture d'un octet sur le port I. I doit être compris entre 0 et 255.

Exemple : PRINT INP(&H8F)

Il s'agit donc de la complémentaire à OUT I,J

****

48.- FONCTION INT

Obtient le plus grand entier inférieur ou égal à X

```
Exemple : PRINT INT(99.99)
          99
          PRINT INT(-32.12)
          -33
```

****

49.- FONCTION LEFT$

A partir d'une chaîne de caractères X$, sélectionne I caractère de gauche. Il ne doit pas dépasser 255. Si I est plus grand que la longueur de XS, toute la chaîne est prise.

```
Exemple : 10 A$ = "TODAY"
          20 PRINT LEFT$(A$,3)
          RUN
          TOD
```

****

50.- FONCTION LEN

Obtient la longueur de la chaîne X$.

```
Exemple : 10 D$ = "LES QUATRE SAISONS"
          20 PRINT LEN(D$)
          RUN
          18
```

****

51.- FONCTION LOG

Obtient le logarithme népérien de l'argument. Cet argument doit être positif.

```
Exemple : PRINT LOG(5.4)
          1.68639895
```

52.- FONCTION LPOS

Obtient la position de tête du pointeur de ligne dans le tampon de sortie de l'imprimante.

```
Exemple : 10 PRINT "DIMANCHE"
          20 PRINT LPOS(X)
          RUN
          8
```

****

53.- FONCTION MID$

Obtient dans la chaîne X$ une sous-chaîne qui commence au caractère de rang I et de longueur J. I et J doivent être compris entre 0 et 255.

Si J est absent, la sous-chaîne va jusqu'à la fin de X$. Si I LEN est trop grand, la chaîne sera vide.

```
Exemple : 10 A$ = "BONJOUR"
          20 PRINT MID$(A$,2,4)
          RUN
            NJOU
```

****

54.- FONCTION PEEK

obtient la lecture de l'octet à la position I de la mémoire (octet entier de 0 à 255).

```
Exemple : A=PEEK(&H2A00)
```

****

55.- FONCTION POINT

obtient le numéro de la couleur sélectionnée par les coordonnées (I,J).

```
Exemple : 10 PSET(7,30),3
          20 PRINT POINT(7,30)
          RUN
          3
```

****

56.- FONCTION POS

Obtient la position de l'axe X du curseur

```
Exemple : 10 LOCATE 12, 7
          20 PRINT POS(X)
          RUN
          12
```


57.- FONCTION RIGHT$

Obtient une chaîne formée des I Caractères de droite X$. Si I=0, la chaîne est vide

```
Exemple : 10 A$="ABCDEFG"
          20 PRINT RIGHT$(A$,4)
          RUN
          DEFG
```

****

58.- FONCTION RND

Syntaxe : RND(x)

Initialiser le générateur de nombres aléatoires Si l'expression facultative manque, l'exécution est suspendue.

Si l'on utilise RND sans initialiser le générateur à chaque exécution du programme, la même suite de nombres aléatoires sera produite.

```
Exemple : 10 FOR I=1 TO 5
          20 PRINT INT(RND(1.0)*100)
          30 NEXT I
          RUN
```

x>0 commence une nouvelle séquence
x=0 donne le dernier nombre généré
x<0 génère un nouveau nomble aléatoire

****

59.- FONCTION SGN(X)

Distinguer le signe de X

Lorsque X.0, alors 1 est donné même `pour 0, si X 0, alors -1 est donné

```
Exemple : PRINT SGN(912)
          1
```

****

60.- FONCTION SIN

Calcule le sinus de X en radians. le résultat est obtenu en précision simple.

```
Exemple : PRINT SIN(0.32)
         .314566561
```

61.- FONCTION SPC

Envoie X nombre d'espaces après le curseur sur l'écran ou l'imprimante.

Utilisation uniquement en PRINT ou LPRINT

Exemple :
```
10 PRINT"GAME"
20 PRINT SPC(3)
30 PRINT"OVER"
RUN
GAME   OVER
```

****

62.- FONCTION SQR

Obtient la racine carrée de X
X doit être zéro ou positif

Exemple :
```
PRINT SQR (10)
3.16227766
```

****

63.- FONCTION STICK

Objet : Donne le contrôle par les poignées de jeu (JOYSTICKS)

Si I désigne le numéro de "JOYSTICK"
I=0 " les touches du clavier
I=1 " le JOYSTICK 1
I=2 " le JOYSTICK 2

Les directions du JOYSTICK sont données par les chiffres sur le dessin figurant à droite.

Exemple 1 : (le manche à balai est connecté sur la prise 1 du synthétiseur)

```
10 IF STICK (1)=1 then print "nord"
20 IF STICK (1)=2 then print "nord-est"
30 IF STICK (1)=3 then print "est"
40 IF STICK (1)=4 then print "sud-est"
50 IF STICK (1)=5 then print "sud"
60 IF STICK (1)=6 then print "sud-ouest"
70 IF STICK (1)=7 then print "ouest"
80 IF STICK (1)=8 then print "nord-ouest"
90 IF STICK (1)=0 then print "manette inactive"
```

La manette de jeu indiquera les points cardinaux.

Exemple 2 : (la manette de jeu est connectée sur la prise 1 du synthétiseur)

On dessine à l'écran un dessin en controlant la manette de jeu.

```
   3 SCREEN 3,1,1
   5 COLOR 3,1,1
   5 CLS
  20 L=81
  30 C=128
  40 PSET(C,L),3
  50 IF STICK (1)=1 THEN L=L-1 : GOTO40
  55 IF STICK (1)=2 THEN 1000
  60 IF STICK (1)=3 THEN C=C+1 : GOTO40
  65 IF STICK (1)=4 THEN 1010
  70 IF STICK (1)=5 THEN L=L+1 : GOTO40
  75 IF STICK (1)=6 THEN 1020
  80 IF STICK (1)=7 THEN C=C-1 : GOTO40
  85 IF STICK (1)=8 THEN 1030
  90 GOTO50
1000 L=L-1
1001 C=C+1
1002 GOTO40
1010 L=L+1
1011 C=C+1
1012 GOTO40
1020 L=L+1
1021 L=C-1
1022 GOTO40
1030 L=L-1
1031 L=C-1
1032 GOTO40
```

****

64.- FONCTION STRIG

Obtient le contrôle du bouton de commande du JOYSTICK (manche à balai)

I désigne le numéro pour le JOYSTICK
I=0 barre d'espacement
I=1 bouton 1 du JOYSTICK
I=2 bouton 2 du JOYSTICK

Lorsque l'on appuie sur la gâchette, c'est le 1 qui sort, autrement c'est le 0.

Exemple :
```
10 IF STRIG (0) = 0 THEN GOTO 10
```

Le programme attend à la ligne 10 le déclenchement d'une touche du clavier.

65.-FONCTION STR$(X)

On obtient une chaîne de caractères qui exprime la valeur de X

```
Exemple : 10 print"contenu ="+STR$(10-90)

          RUN
          contenu =-80
```

(voir aussi l'instruction VAL)

****

66.-FONCTION TAB(X)

On obtient X espaces sur l'écran jusqu'a ce que le curseur soit à la position X

```
Exemple : 10 PRINT"XXX";TAB(10);"XXX"
          20 PRINT"PHC-25";TAB(10);"PROGRAMME"
          RUN
          XXX          XXX
          PHC-25       PROGRAMME
```

****

67.-FONCTION TAN(X)

On obtient en simple precision la tangente de X exprimée en radians

```
Exemple: PRINT TAN(1.34)
         4.67344123
```

****

68.- FONCTION TIME

Obtient un compteur de temps qui ajoute 1 chaque 1/360 secondes.

```
Exemple : 10 A=TIME
          20 B=TIME-A
          30 PRINT INT(B/360)
          40 GOTO 20
```

N.B. Il ne peut y avoir de grande précision dans ce compteur.

****

69.- FONCTION USR

Syntaxe USR(X)

Exécute un programme fait par l'utilisateur en langage machine.

(voir instruction EXEC)

****

70.- FONCTION VAL

Syntaxe : VAL(X$)

Obtient la valeur de la chaine de caractères X$

Si le premier caractère de X$ n'est pas +,-,&/ou un chiffre, la valeur devient 0.

Si un caractère autre qu'un nombre (en notation hexadécimale 0 - 9, A - F) apparaît, les caractères suivants sont ignorés.

```
Exemple : 10 FOR I=1TO 3
          20 READ A$
          30 PRINT VAL("&H"+A$);
          40 NEXT
          50 DATA C3,40,FF
          RUN
           195 64 255
```

71.- FONCTION SCRIN

Syntaxe SCRIN(X,Y)

Renvoie la valeur ASCII du caractère de coordonnées d'écran x,y

x étant le numéro de la colonne (la première étant 0)
y étant le numéro de la ligne (" ")

Exemple :
```
5 CLS
10 PRINT"ABCDEF"
20 Z=SCRIN(5,0)
30 PRINT Z

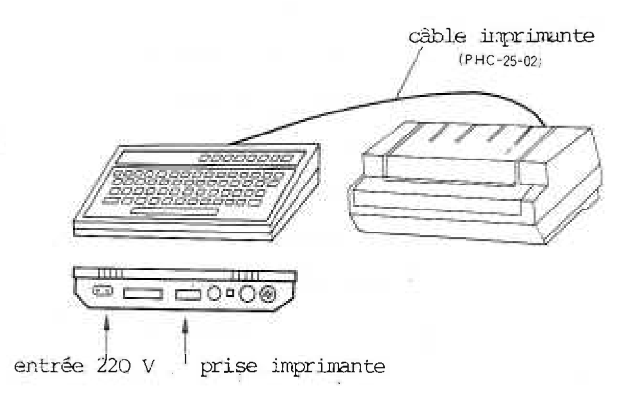
RUN
ABCDEF
70
```

Commentaires : 70 est la valeur ASCII en décimal de F, colonne numéro 5 de la ligne numéro 0.

ANNEXE A CONNECTION AVEC L'IMPRIMANTE

Utiliser le câble spécial vendu en option (PHC 25-02)

pour connecter l'imprimante au connecteur prévu à l'arrière de l'appareil.



a)DATA STROBE

Il s'agit du signal de sortie fait vers l'imprimante et utilisé comme synchronisation lorsque les données sont envoyées vers l'imprimante.

b) DATA - 07 -

Il s'agit de la sortie des données du bus 8 bit parallèle envoyé vers l'imprimante.

c) READY

Le signal READY permet l'emission des instructions. Lorsque ce signal est à un niveau bas, il est possible d'envoyer des data vers l'imprimante.

d) GND (GROUND)

Prise de masse donnant le potentiel à tous les signaux.

CONFIGURATION DU CABLE IMPRIMANTE

DATA STROBE	1	1	DATA STROBE
DATA 0	2	2	DATA 0
DATA 1	3	3	DATA 1
DATA 2	4	4	DATA 2
DATA 3	5	5	DATA 3
DATA 4	6	6	DATA 4
DATA 5	7	7	DATA 5
DATA 6	8	8	DATA 6
DATA 7	9	9	DATA 7
	10	10	ACKNOWLEDGE
READY	11	11	READY
	12	12	
GND	13	13	
GND	14	14	
		15	
		16	
		17	GND
		18	
		19	GND
		36	

connecteur côté SANYO 14 pins — connecteur côté imprimante 36 pins

ANNEXE B FONCTION DES TOUCHES DE CONTROLE

Contrôle C : l'éxécution du programme est arrêtée

" G : avertissement sonore (Buzzer) avec synthétiseur.

" I : tabulation - le curseur bouge ver la droite par groupe de 8 colonnes

" J : entrée de ligne

" L : effacement de l'écran

" M : retour de chariot

" Q : changement de page sur l'écran

" ESC : arrêt temporaire du programme
pour remettre le programme en marche
on peut appuyer sur n'importe quelle
touche

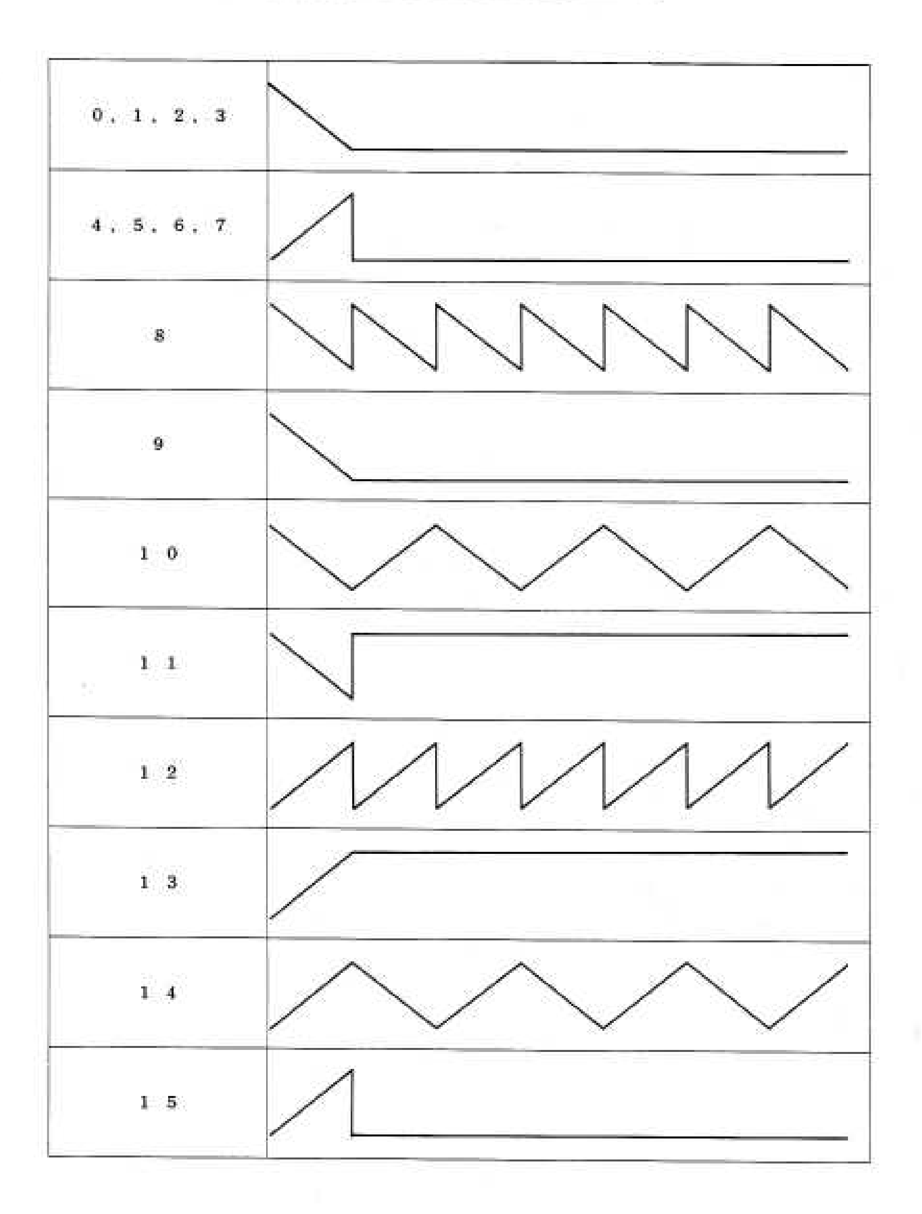
ANNEXE C LISTE DES NOMS VARIABLES INTERDITS

ABS	ASC	CHR$
CLEAR	CLOAD	CLOAD?
CLS	COLOR	CONSOLE
CONT	CSAVE	CSRLIN
CTON	CTOFF	
DATA	DEFFN	DIM
END	ELSE	EXEC
EXP	FOR	FRE
GOSUB	GOTO	IF
INKEY$	INP	INPUT
INT	KEY	LCOPY
LEFT	LEN	LET
LINE	LIST	LLIST
LOCATE	LOG	LPOS
LPRINT	MID$	NEXT
NEW	ON	OUT
PAINT	PEEK	PLAY
POINT	POKE	POS
PRESET	PRINT	PSET
READ	REM	
RESTORE	RETURN	RIGHT$
RND	RUN	SCREEN
SGN	SIN	SOUND
SPC	SQR	SCRIN
SLOAD	SSAVE	STEP
STICK	STOP	STRIG
STR$	TAB	TAN
THEN	TIME	TO
USR	VAL	FN

ANNEXE D SOLUTION AUX PROBLEMES

ERREUR	VERIFICATION A FAIRE
L'écran n'affiche pas	1. L'alimentation est-elle branchée sur le PHC 25 ? 2. L'alimentation est-elle branchée sur le moniteur vidéo ou poste TV? 3. La connection entre le PHC 25 et le moniteur vidéo SG 12 est-elle en place? 4. La connection entre le PHC 25 et l'adaptateur pour le poste TV (lorsque c'est le cas) est-elle faite? 5. Contraste ou luminosité du poste moniteur ou TV
L'écran n'est pas stable	Effectuer les règlages verticaux et horizontaux de synchronisation
Arrêt en cours d'émission	1. S'agit-il d'une coupure de courant? 2. Interférence d'un autre équipement. Evitez l'utilisation d'un poste de radio etc... 3. Vérifiez votre programme
La commande BREAK n'a pas d'effet	Fermez l'interrupteur et recommencez à nouveau
CSAVE ou CLOAD ne fonctionnent pas	1. Recommencez toutes les séquences en suivant rigoureusement les indications du mode d'emploi. 2. Vérifiez le magnétophone et l'état du chargement des piles si vous utilisez des piles. 3. Vérifiez l'adaptateur et l'alimentation du magnétophone. 4. Vérifiez le règlage de volume et de tonalité. 5. Nettoyez la tête de lecture

FORME DES SIGNAUX GENERES PAR LE SYNTHETISEUR MUSICAL



CONTENU DU GENERATEUR DE CARACTERES

Lower 4 bits \ Upper 4 bits	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		π		0	@	P		p	♠							
1		┴	!	1	A	Q	a	q	♥							
2		┬	"	2	B	R	b	r	♣							
3		┤	#	3	C	S	c	s	♦							
4		├	$	4	D	T	d	t	○							
5		┼	%	5	E	U	e	u	●							
6		│	&	6	F	V	f	v								
7		─	'	7	G	W	g	w								
8		┌	(	8	H	X	h	x								
9		┐	)	9	I	Y	i	y								
A		└	*	:	J	Z	j	z								
B		┘	+	;	K	[	k	{								
C		×	,	<	L	\	l	|								
D			−	=	M	]	m	}								
E			.	>	N	^	n	~								
F			/	?	O	_	o									

MEMOIRE

Zone	Adresse
ZONE DE TRAVAIL DE BASIC	FFFH
PAGE VIDEO 2	F800H
ZONE DE TRAVAIL DU PROGRAMME	E000H
ESPACE LIBRE	C000H
PAGE VIDEO 1	8000H
PAGE VIDEO 1	7800H
INTERPRETEUR BASIC	6000H
	0000H

LISTE DES MESSAGES D'ERREUR

Bad subscript

- Les valeurs désignées par des indices sont-elles en dehors des dimensions du tableau déclaré par DIM ?
- les nombres sont-ils différents des nombres déclarés par DIM ?

Can't continue

- La commande CONT ne peut pas être utilisée après les corrections d'un programme.
- Y a-t-il des erreurs dans le programme ?

Duplicate definition

- Y a-t-il une déclaration DIM en cours utilisant un nom qui fait double emploi ?

Illegal function call

- Le nombre utilisé est-il en dehors des paramètres autorisés ?

Illegal direct call

- Une commande incorrecte a été faite. Les instructions DEFFN sont-elles utilisées en direct mode alors que ce n'est pas autorisé ?

Too long string

- La longueur de la chaine de caractères dépasse-t-elle 255 caractères ? Vérifiez à l'endroit où la chaîne de caractères est additionnée.

Missing operand

- Une operande indispensable manque. Le côté droit de = commande de substitution est-il en place ?
- Les opérateurs sont présents, mais les valeurs et caractères sont-ils correctement inscrits ?

Next w/o for

- S'agit-il d'une instruction NEXT sans l'instruction FOR?
- Vérifiez la boucle FOR - NEXT.

Not enough data pas assez d'information

- Lorsque l'ordinateur procède à la lecture, il n'y a pas de data.
- Vérifiez le nombre de data dans l'instruction DATA et le nombre de variables dans l'instruction READ.

Out of memory

- Le programme est trop grand, il dépasse la mémoire. Lorsque le programme est long, il faut toujours vérifier la mémoire disponible en faisant appel à la fonction FRE (X).

Our of string space

- La chaîne de caractères est insuffisante. Augmenter la zone de la chaîne de caractères en utilisant la commande CLEAR.

Overflow Dépassement de capacité

- Le résultat du calcul est supérieur à 1.7 E 38.
- Le PHC 25 est limité à 1.7 E 38.

Return w/o gosub

- S'agit-il d'une instruction RETURN sans instruction GOSUB?
- Vérifiez la correspondance des deux instructions.

Syntax error

- La frappe a-t-elle été faite correctement ?
- Les noms variables sont-ils corrects ?

String too complex

- La chaîne de caractères est trop compliquée.

Type mismatch

- Les chaînes de caractères et les valeurs numériques sont-elles mélangées ?

Tape read error

- La cassette n'a pas été lue correctement.

Underfined FN call

- S'agit-il d'une utilisation d'une fonction FN qui n'est pas définie par l'instruction DEFFN ?

Underfined line number

- Le numéro de ligne désigné en GOTO, GOSUB, instruction RESTORE ou commande RUN n'existe pas.

Divided by zéro

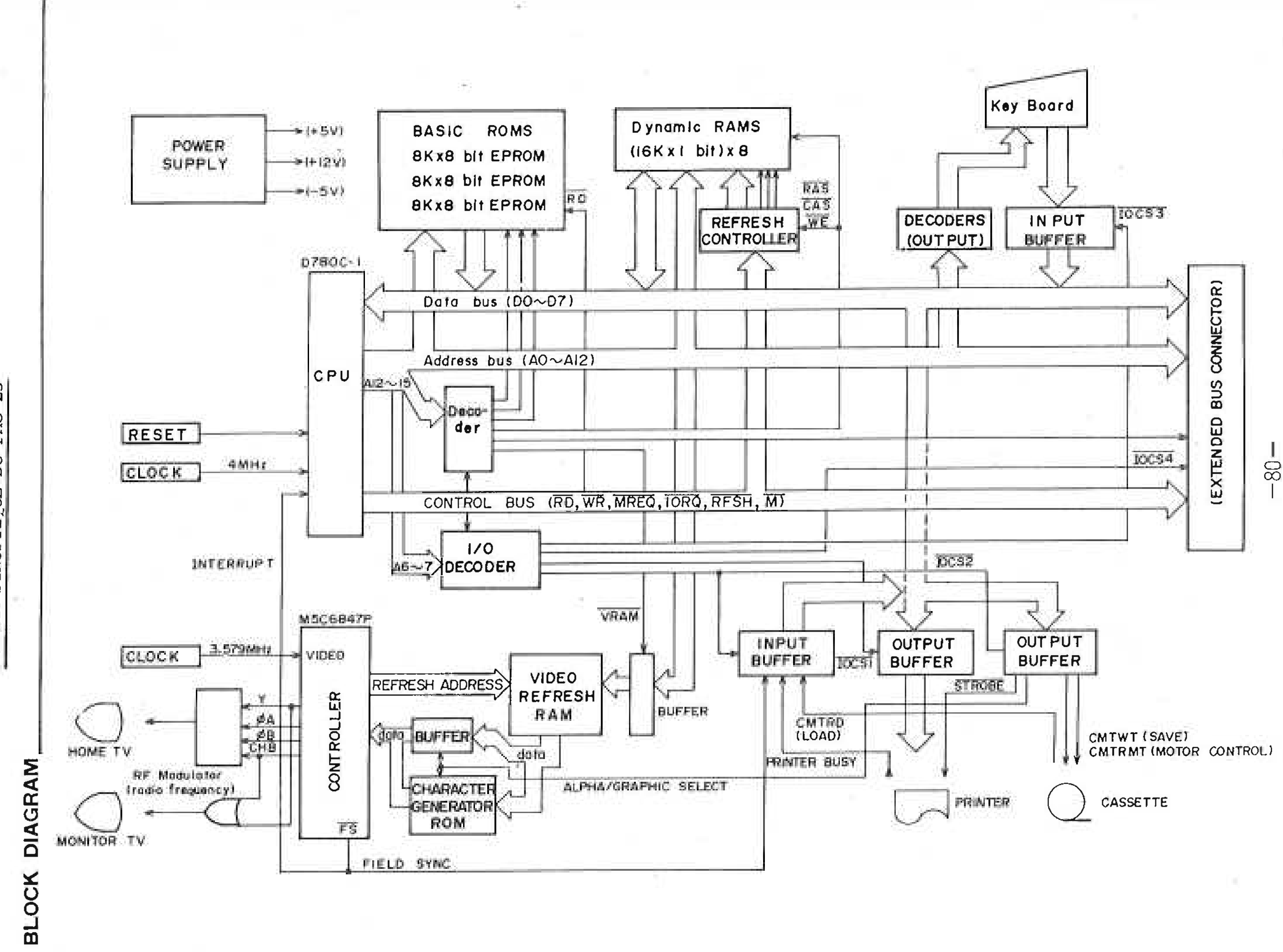
Une division par zéro a été tentée, ce qui est impossible.

Le PHC 25 considère toute valeur inférieure à 1×10^{-38} comme zéro.

============

SCHEMA SYNOPTIQUE DU PHC 25

BLOCK DIAGRAM



CONFIGURATION DES CONNECTEURS

CASSETTE CONNECTOR

(HOSIDEN TCS4480)

PIN NO.	SIGNAL	
1	GND	
2	GND	
3	GND	
4	CMTOUT	Red
5	CMTIN	White
6	REM+	Black
7	REM−	
8	GND	

PRINTER CONNECTOR

PIN NO.	SIGNAL
1	DATA STROBE
2	DATA0
3	DATA1
4	DATA2
5	DATA3
6	DATA4
7	DATA5
8	DATA6
9	DATA7
10	
11	READY (PBUSY)
12	
13	GND
14	GND

21 PIN CONNECTOR

PIN NO.	SIGNAL
1	BLUE
2	SELECT
3	GND
4	RED
5	RF VIDEO
6	GREEN
7	+12V
8	VIDEO

EXTENDED CONNECTOR

PIN NO.	SIGNAL	PIN NO.	SIGNAL
1	D0	2	GND
3	D2	4	D1
5	ROMRAMCS	6	D3
7	A12	8	WR
9	A10	10	A11
11	A6	12	A9
13	A8	14	A7
15	A4	16	A5
17	A2	18	A3
19	A0	20	A1
21	IOCS4	22	CLOCK φ 4MHz
23	D6	24	D7
25	D4	26	D5
27	IORQ	28	RD
29	RESET	30	M1
31	Vcc	32	Vcc
33	GND	34	GND

MEMORY MAP

0 0 0 0	BASIC-ROM
1	
2	
3	
4	
5	
6	VIDEO RAM
7	
8 0 0 0	(extended ROM.RAM area)
9	
A	
B	
C 0 0 0	(standard RAM area)
D	
E	
F	

I/O MAP

Chip Select	A_7	A_6	A_5 A_4 A_3 A_2 A_1 A_0	R/W	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	Note
IOCS1	0	0	× × × × × ×	W	PD_0	PD_1	PD_2	PD_3	PD_4	PD_5	PD_6	PD_7	printer out put DATA
IOCS2	0	0	× × × × × ×	R					FS	CMTRD	PBUSY	HS	
				W	CMTWT	CMTRMT	カナ	PSTRB	GMI	GMO	CSS	A G	
IOCS3	1	0	× × × × × ×	R	KD_0	KD_1	KD_2	KD_3	KD_4	KD_5	KD_6	KD_7	KEYBOARD INPUT DATA
IOCS4	1	1	× × × × × ×										EXTENDED I/F

Table of Character Code

Lower 4 bits \ Upper 4 bits	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		•		0	@	P	`	p	♠							
1			!	1	A	Q	a	q	♥							
2			"	2	B	R	b	r	♣							
3			#	3	C	S	c	s	♦							
4			$	4	D	T	d	t	○							
5			%	5	E	U	e	u	●							
6			&	6	F	V	f	v								
7			'	7	G	W	g	w								
8			(	8	H	X	h	x								
9			)	9	I	Y	i	y								
A			*	:	J	Z	j	z								
B			+	;	K	[	k	{								
C		X	,	<	L	\	l	|								
D			-	=	M	]	m	}								
E				>	N	^	n	~								
F			/	?	O	—	o									

Note: Hexadecimal notation — Decimal notation

Hexadecimal notation		Decimal notation
&H00	=	0
&H09	=	9
&H0A	=	10
&H0C	=	12
&H0F	=	15
&H10	=	16
&H11	=	17
&H1F	=	31
&H20	=	32
&HFF	=	255

LISTE DES INSTRUCTIONS DU BASIC ETENDU DU PHC 25

LISTE DES INSTRUCTIONS DU BASIC ETENDU DU PHC 25

CET APPAREIL EST CONFORME A L'ARRETE DU 14 JANVIER 1980 (J.O. DU 17 JANVIER 1980) RELATIF A LA LIMITATION DES PERTURBATIONS RADIO-ELECTRIQUES.



SANYO ELECTRIC CO., LTD.
OSAKA, JAPON

R-4777427